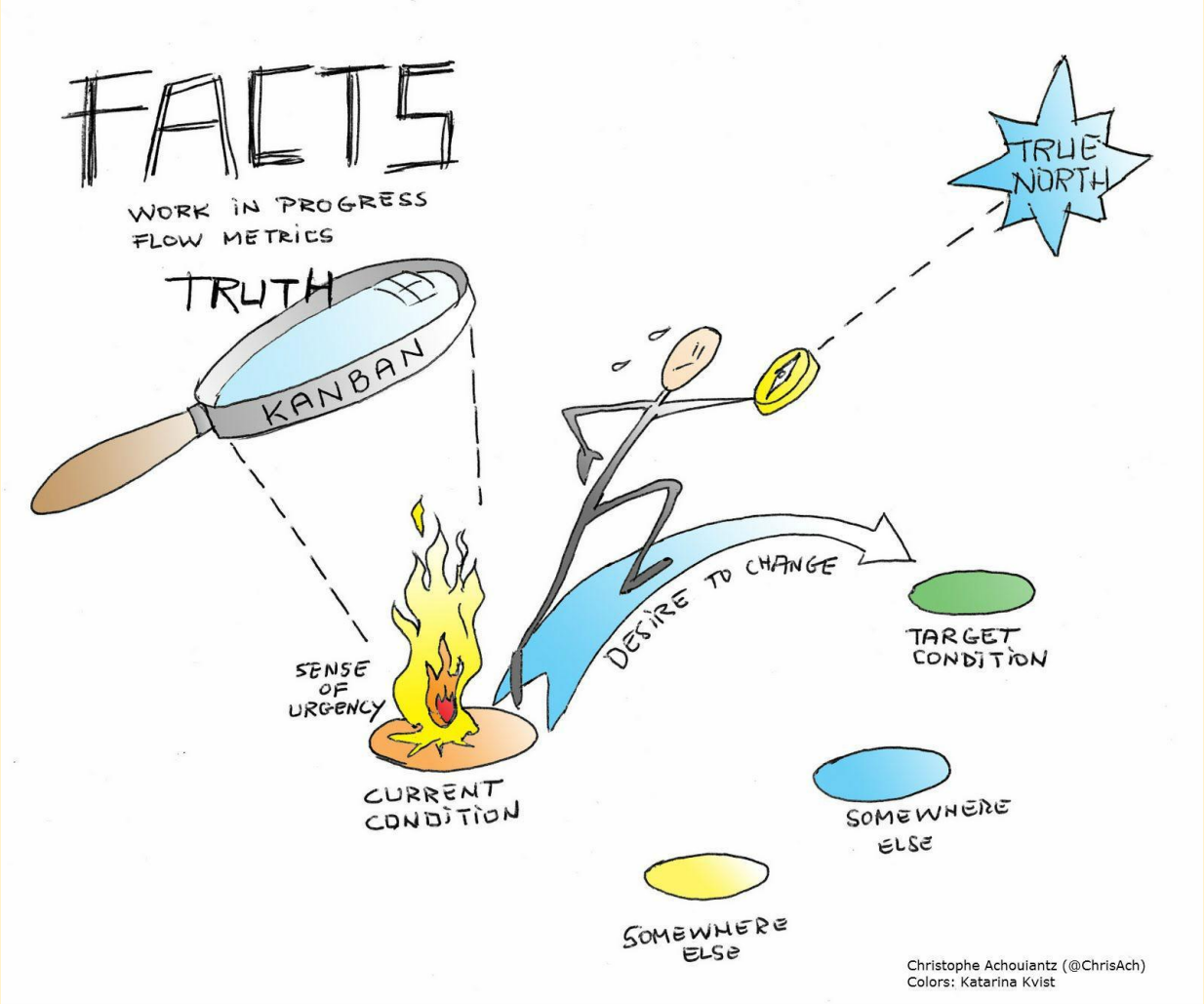
The Kanban Kick-start Field Guide

Create the Capability to Evolve



Sandvik IT

Version 1.1 (20131125)

Christophe Achouiantz - Lean/Agile Coach - Sogeti (@ChrisAch)

Johan Nordin – Lean Coach – Simplify Change (@JohanNordin)

Introduction

This field guide is about how to introduce a team to the Kanban method. Therefore, its scope is wider than just introducing a kanban technique; it really is about injecting the capability to evolve in a continuous and sustainable way. It is about discovering a way of working with attached policies that make sense in the team's context in order to deliver the right thing, just in time. It is about the team doing the smartest thing and becoming what it *must* to succeed. It is quite simply about becoming Lean.

Why a field guide?

Kanban is spreading fast within Sandvik IT, to the extent that the coaches from Operational Excellence Support cannot (and shall not) conduct all implementations. This guide is intended for managers, flow managers and coaches within Sandvik IT that want to introduce Kanban to their team(s). It is a repository of best practices to introduce Kanban based on the experience gathered kick-starting 50+ teams at Sandvik IT between December 2010 and Mars 2013.

What's in this Guide?

This field guide will help you prepare, create, run and follow-up a one day workshop to kick-start a team into using the Kanban method. The focus of the guide is on evolutionary changes (improvements) rather than on pure flow improvement. The reader - the "coach" - is meant to be the person preparing, facilitating and following-up the workshop.

How to Use this Guide?

The guide has two parts: the first part presents the Kick-start concept and summarizes its features. The second part is meant to be used as a reference on how to run specific features by providing details, tips and comments.

Scope of the guide

This guide is not an introduction to the Kanban method. It requires the reader to be well familiarized with Kanban. For more information regarding Kanban, please read David J. Anderson's book "Kanban: Successful Evolutionary Change for Your Technology Business".
Because it is a field guide, this document is very practical and focuses on *what* to do and *how* during the workshop. There is very little background or theory, and we try to keep the language simple and not overloaded with expert jargon. That been said, we expect the reader to have basic Kanban and Agile knowledge (e.g. knowing about retrospective, Definition of Done, class-of-services, etc.).

A Current Standard

This guide contains only practices and recommendations that have been proven in the field many times and that are part of the current standard we use within Sandvik IT. You will not find any unproven ideas, cool-things to try or R&D here. We have of course plenty of those, but you will not find them here - yet; they may well make it into this guide once they are validated in the field. As a result, this guide is in constant and rapid evolution, and the standard is simply the current way we "do things". We expect to update it frequently based on our own experience and, if you want, your feedback.

Disclaimer for Usage Outside of Sandvik IT

The setup described in **this guide is not THE way to implement Kanban, it is but A way that works in the context of Sandvik IT**. That which works for us will not necessarily work for you. As your context is not Sandvik IT, you should not follow this guide blindly. Think and adapt the setup described in this guide to fit your situation.

There are many other ways to introduce Kanban to a team, like David J. Anderson "Systems Thinking Approach to Introducing Kanban", which may have other approaches and goals. For instance, the approach described here is very much for the team and by the team.

Licensing

About Sandvik IT

Sandvik, founded in 1862, is a high-technology, engineering group. The Sandvik Group conducts operations in five business areas: mining, machining solutions, materials technology, construction and venture. Worldwide business activities are conducted through representation in more than 130 countries. In 2012 the Group had about 49,000 employees with annual sales of approximately 11,880 MEUR.
Sandvik IT is supporting the Sandvik group's IT needs. In 2012, Sandvik IT has about 900 employees in 40 countries, supporting 365 applications for more than 30.000 users.

Table of Content

The Kick-start Concept

What is the Kanban Kick-start?

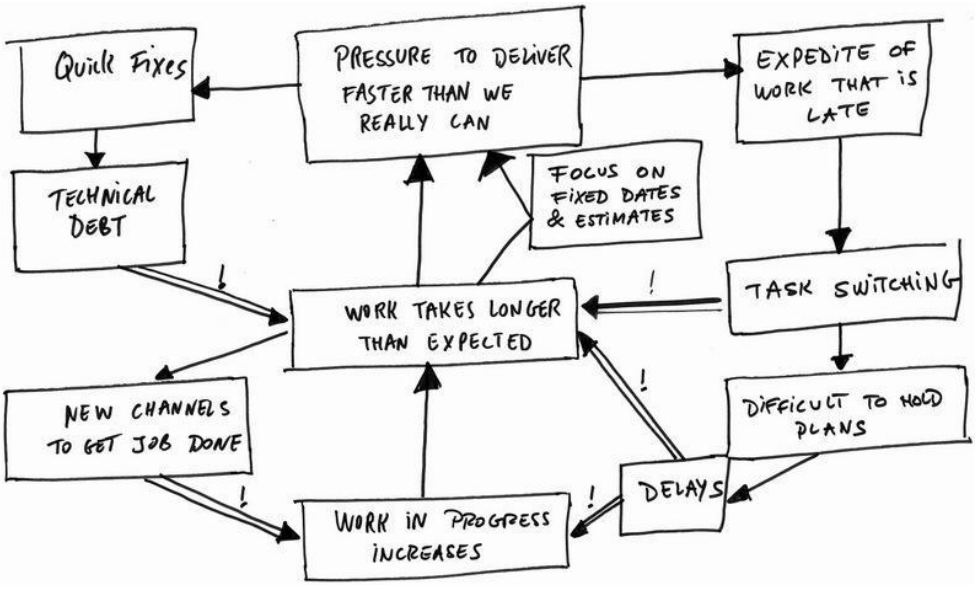
The Kanban Kick-start is a one day workshop whose goal is to help a team putting in place a kanban system. It is packaged as a one-day event to be easily “sold” to stressed managers and teams.

The workshop engages the team members to discuss and agree on what the team actually does, why, how and for whom. Based on this understanding, the team can create *visualization* policies to see the work in progress (usually using stickies on a board) and *way-of-working* policies that govern how the work works. The workshop also set in place policies to *act* on this new understanding by planning the smartest way ahead using planning meetings as often as needed (usually daily).

Basically, the workshop helps the team to get in control of its workflow. This is the basis for improvement for a team. Without the control and understanding given by the kanban system there cannot be long-lasting improvements. That is the meaning of the Kick-start.

History of the Kanban Kick-start at Sandvik IT

It didn’t start with Kanban, it started with the realization that things were not as good as they should. Employees were frustrated and customers unsatisfied: they experienced increased technical depth, difficulties to predict when and what to deliver and unhappiness about their current work situation. We could see stressed people running and switching between tasks, feeling pressure to deliver something that no one was proud of, frustrated by the lack of control over what was happening. Analyzing these pains, we identified that the one reoccurring root-cause for all these problems was the fact that the teams were not in control of their flow of work.



There was an interesting catch 22 situation here: the teams needed to control their flow of work, though without control of the workflow the teams simply didn’t have time to let us help them; there was way too much stress and pressure to deliver the thousands of urgent things that were going on. So, what could we do? Perhaps we could find a way to help these teams in small steps; perhaps they would have just enough time for that. It would require the teams to see and understand their current pains in order to address them in the right order, in baby steps, one at a time, slowly improving. We didn’t need a revolution; we needed to introduce an evolutionary capability into those teams.

This very problem is exactly what David J. Anderson addresses in his book "Kanban: Successful Evolutionary Change for Your Technology Business". So, the Kanban method became the *tool* we needed to build the capability to continuously improve our teams. Let's use Kanban!

But, wait! Our challenge quickly became: how to introduce Kanban to all these teams with our limited resources? Introducing a kanban system is a change in itself and requires the team to grasp some concepts that at first may be alien (What do you mean "pull"? You mean that I cannot pick a new card because some guy decided that we are limited to 10 cards?). Moreover, we had to ask the team managers to give us some precious time to start a kanban system, time that was not - at least short term - contributing to solving the customer's problems. How much time did we need? The managers were adamant: we could spare a day, no more; you will have to do with it!

Our solution has then been to - within one day - help a team reach a "good enough" control over their workflow so that they can start improve, at their own pace. The challenge is then to - within this one day - be able to take any team to a state where it's members understand the team's purpose, visualize the ongoing work and current workflow, discuss and agree on their most important and relevant work policies and, using all of that, to be ready to start planning their daily work the very next morning.

We call it the Kanban Kick-start.

For more information about our journey, please check this recording of a session we did at the Lean Kanban Central Europe 2011 conference, entitled "Igniting change in 20 teams within 6 months" (http://vimeo.com/57062630).

Key Aspects

Due to our *constraints* (few coaches for many teams, focus on evolution rather than direct better flow) the Kanban Kick-start has developed the following aspects that may differ from other ways/methods/processes for introducing Kanban:

- The team is the driver. The coach is simply helping the team to set its kanban system in place.
- In its purest approach, the coach does not design the kanban system for the team, the team does it. This may not become an optimal kanban system, but this is not the goal.
- The goal is to increase the team's understanding about itself and its context so that it can **evolve by itself** and **become what it must**.
- At the end of the workshop, the team has a visualization board, explicit work policies and a way-of-working with the board.
- The workshop, the board and the policies are for the team and by the team.
- The agenda of the Kick-start is modular to fit teams with different maturity.
- The Kanban Kick-start is packaged in such a way to be easily "sold" to managers under stress: It is a one day workshop with the whole team (6-8 hours).
- Not everything actually happens during one day, some preparations are required before the workshop with the manager and flow manager.
- The coach is also required to follow-up the team after the workshop to: a) help the team implement the Kanban concepts that didn't stick during the workshop (or that were left out), and b) make sure that the team can sustain its kanban system (by adapting the system or coaching the team).

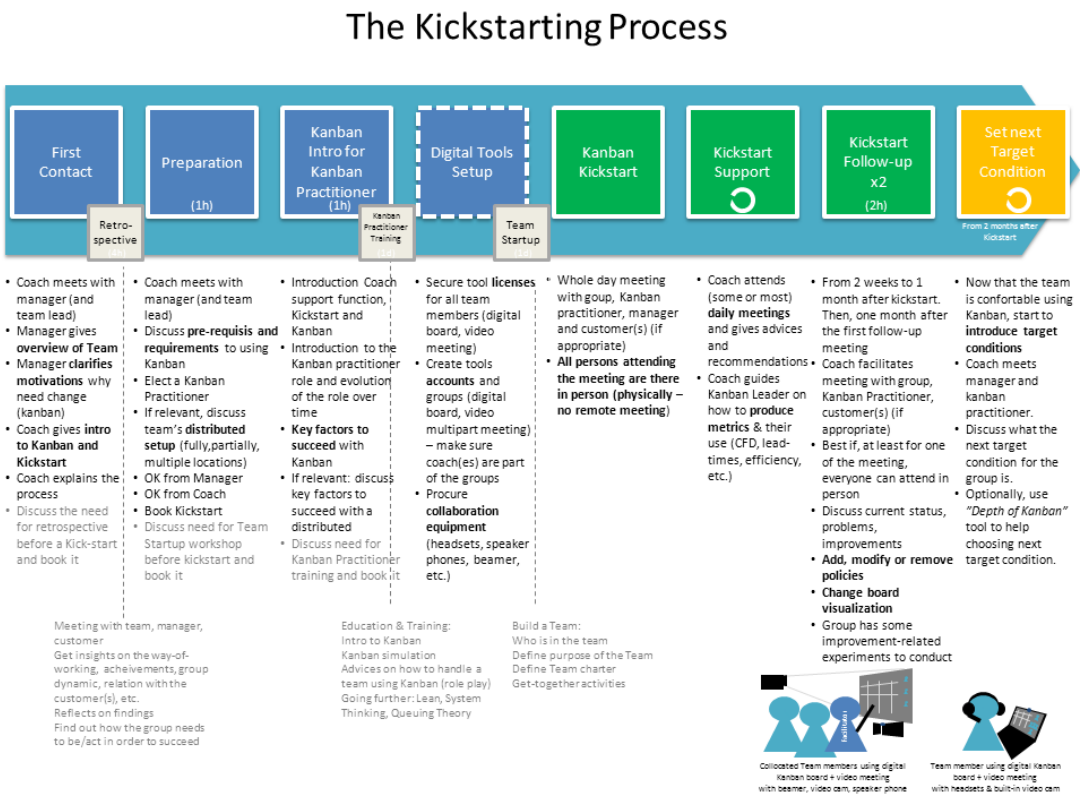
A Process for Evolution

As described earlier, the Kanban Kick-start is the one day workshop used to kick-start a team in using the Kanban method. But by extension it also refers to the *process* to get a team from the state it is in to a point where it can sustainably and continuously improve itself: a kick-starting process if you will.

This guide describes the whole kick-starting process, and not only the actual workshop. You will find support in this guide related to the preparations needed for introducing the Kanban method, the actual Kick-start workshop, and the support required after the Kick-start in order for the team to reach a point where it can work towards target conditions.

A Kick-starting Process

Based on the content of this guide, here is an example of a kick-start process used at Sandvik IT. Your process can look more or less complex depending on how many features you want to cover.



The Goal: Take control so that you can start improving

The goal of the kick-starting process is for the team to *take control* over their current way of working and reach a stable *continuous improvement baseline.* This baseline is the point where the team is truly aware of its current condition and has a good foundation to start improving upon. Of course, a team doesn't need to be "at baseline" to work with improvements, though what we really are after is for the team to improve continuously, in a *repeatable* and *sustainable* way, and that the results can be sustained *forever*. This requires a specific condition for the team: the baseline. Without that baseline, any improvement work will simply not stick and will end-up as "something we tried that didn't work".
The Kick-start is currently the fastest way we know of to bring a team to continuous improvement baseline.

A team is considered to be at the *continuous improvement baseline* when its team members can:

- Understand the team's current condition using leading indicators (visualization elements).
- Understand the team's current capability using lagging indicators – lead-times (see Appendix I: Definition of Lead-times), throughput, defect generation rate, due-date performance, etc.
- Follow a common way-of-working using explicit policies (note that these policies do not have to be perfect nor "right" - yet),
- Demonstrate the ability to maintain these explicit way-of-working policies, as well as modify them when necessary (add, remove, modify)

Once the team is at baseline, it can start its real journey: *improving its way of working*. Another way to look at the continuous improvement baseline is that it is the first (mandatory) target condition we set to a team.

The scope of this field guide is to help the team reach this baseline. What happens after that is very much depending on the team's current situation and the organization's goals. The "After the Kick-start" section of this guide gives some hints on how to go further using Toyota Kata's improvement and coaching katas.

The Kick-start

Here is a quick summary of the different “features” available to you when you want to kick-start a team. The features are arranged into steps: before the kick-start, running the kick-start and after the kick-start. Check the “How To” section of this guide for more details about each feature.

Before the Kick-start: Understand the Team

Definition of Done:

- *Someone (team manager, team-lead, or any other team member) is ready to act as Flow Manager.*
- *The manager and flow manager know what Kanban is about.*
- *You understand the team’s context.*
- *You are confident that the necessary preconditions are fulfilled for the team to implement and sustain Kanban.*
- *The expectations are set for the manager, flow manager and coach.*
- *GO or No GO from the team manager*

PREP 1: Present Kanban as a Tool for Evolution
Present Kanban to the team manager and flow manager.

PREP 2: Study the Context
Meet the team manager and the flow manager to understand the team and its context.
Purpose | *Context* | *Why Kanban?* | *Acceptance criteria*

PREP 3: Set Expectations
Set expectations with the team manager and flow manager.
Manager | *Flow Manager* | *Coach*

PREP 4: Prepare the Flow Manager
Prepare the flow manager to be ready for the Kick-start
Role | *Role over time* | *Responsibilities*

PREP 5: Get a GO/NO GO Decision
You get a GO/NO GO decision from the Team manager to proceed with the Kanban implementation

Before the Kick-start: Plan the Workshop

Definition of done:

- *You have an agenda for the workshop that fits the team’s current situation.*
- *You have booked the time and place for the workshop and invited the whole team.*
- *You have addressed various practical details for the workshop.*

SETUP 1: Create an Agenda
You build an agenda for the Kick-start workshop that fits the team’s current situation and maturity.
Select features | *Pauses*

SETUP 2: Setup the Workshop
You go through all the little practical details for holding the workshop.
Who attends | *The board* | *Props* | *Room* | *Invitation*

SETUP 3: Extra Preparations for a Distributed Team
You have discussed and prepared the tools necessary for the team to communicate distributed.
Collaboration tools

Run the Kick-start

Definition of Done:

- *The team has a board visualizing the on-going work, the most important work that is ready to be started, the workflow (or way-of-working) for the team, and various policies governing the work.*
- *The team has discussed and agreed on various visualization and work-related policies. These policies describe the team's current way-of-working (no "wanted state" here).*
- *The team has a way to work with the board to plan the team's activities (daily or more if necessary).*

KICK 1: Set the Scene
Why a workshop? What is the goal? What will happen during the workshop?
Welcome | *Why* | *Goals* | *About Kanban Kick-start* | *Golden Rule* | *Agenda*

KICK 2: Share Current Concerns
What issues does the team have now? What issues can be resolved today?
Gather Inputs | *Share* | *Categorize* | *Sort*

KICK 3: Define a Shared Vision
Help the team discuss and agree on its Purpose.
Purpose | *Slogan* | *Criteria for success* | *Team composition*

KICK 4: Discover What the Team Does
Help the team understand what it actually does and for whom.
Upstream | *Downstream* | *Responsibility* | *Workflow* | *Dependencies*

KICK 5: Identify Work Types
Help the team understand its demand on how to categorize it.
Gather Data | *Analyze*

KICK 6: Explain How the Board Works
Help the team understand the mechanics of a Kanban board.
Work Item | *Work Type* | *Work Flow* | *Pull* | *…*

KICK 7: Present a Common Language
Help the team understand and use a common language.

KICK 8: Set Visualization Policies
Help the team set policies on how to visualize work.
Work Types | *Work Items* | *workflow* | *What not to visualize* | *When must be in Sync*

KICK 9: Create & Populate the Board
Help the team create and populate a board, according to the team's policies.
Draw workflow | *Create work items* | *Populate board* | *Reflect* | *DoD* | *Pull* | *Blocks* | *Due Dates* | *Prio* | *DoR*

KICK 10: Set Way-of-Working Policies
Help the team set policies governing the team's work.
What to work on | *Side orders* | *When to pull* | *Part-times* | *DoD* | *DoR* | *Blocks* | *Due Dates* | *Standard work*

KICK 11: Limit WIP
Help the team limit the work in progress.
Uncontrolled Flow | *Value Finishing* | *Strategies for reducing WIP* | *Set explicit WIP limits*

KICK 12: Set Planning Meeting Policies
Help the team set policies governing planning meetings.
The planning meeting | *Meeting policies* | *Policies during the meeting* | *The after meeting*

KICK 13: Run the First Planning Meeting
Help the team run its first planning meeting.

KICK 14: Close the Scene
Summarize what was done during the workshop and get feedback.
What happened | *What is next* | *Loose ends* | *Feedback*

After the Workshop: Reach a Baseline

Definition of Done:

- *The team reached Continuous Improvement Baseline*
  - *The team sustains its current policies.*
  - *The team regularly updates, adds and remove policies to fit its current understanding of its role and context.*
  - *The team has policies to manage its demand.*
  - *The team's planning meetings have the right outcome.*
  - *The team measures its flow.*
  - *The team has a strategy to limit its WIP.*

BOOST 1: Attend Planning Meetings
Help the team conduct planning meetings as intended.
Observe and point-out problems

BOOST 2: Follow-up
Help the team adapt its board and policies to the new understanding.
Build common understanding | *Visualization policies* | *Working policies*

BOOST 3: Understand Current Capability
Help the team understand its capabilities.
CFD | *Control Chart* | *Create understanding* | *Investigate*

BOOST 4: Limit WIP
Help the team limit the work in progress.
Limit WIP

BOOST 6: Set Policies for Demand Management
Help the team select the "right" work items to work on.
Understand demand | *Manage demand*

BOOST 7: Assess Depth of Kanban
Help the team assess the depth of its Kanban implementation.
Explain Kanban | *Assess depth*

BOOST 8: Set Next Target Condition
Help the team set a target condition.
Identify direction | *Identify target condition* | *Understand what is in the way* | *Remove obstacles*

BOOST 9: Give Feedback
Contact us

How-To

Reference

Before the Kick-start: Understand the Team

Somehow, a manager got in contact with you to kick-start a team. It may be that the manager heard success stories from other teams that use Kanban (bottom-up approach) or that Kanban is just the way we do things around here (top-down approach). The reason may be varied (not good enough performance, need control, need to evolve before it's too late, etc.).
Regardless on how the idea initiated, the first thing you must do is to understand the needs of the team to be Kick-started. What is the team's context? What are its challenges? Do you believe you can help the team improve using Kanban?
Once you have sufficient information on the team, you must set expectations right with the team's key players: the team manager and the flow manager. Kanban will bring changes to the team and the manager must be ready for it. Then, and only then, can you get a GO or NO GO decision from the team manager about starting the Kanban implementation.

Definition of Done:

- *Someone (team manager, team-lead or any other team member) is ready to act as Flow Manager to lead the Kanban implementation.*
- *The manager and flow manager know what Kanban is about.*
- *You understand the team's context.*
- *You are confident that the necessary preconditions are fulfilled for the team to implement and sustain Kanban.*
- *The expectations are set for the manager, flow manager and coach.*
- *Go or NO GO decision from the team manager*

Key Players

For the purpose of this guide we introduce four key players. These are meant to clarify who does what when discussion the kick-start. They are not meant as roles you must have in your organization for the kick-start to work.

Team

A team is a *learning network of individuals creating value*. In much simpler words, the team is doing some work that (hopefully) is value adding.

This guide works for all kind of teams, although most of the examples given here are software/system development focused. We have kick-started all sort of teams: management teams (steering committees, department managers), "knowledge work" teams (e.g. communications, HR), coordination teams (e.g. incidents & problems management), project and programs teams, etc. This guide applies to all these different types of teams.

Note that you get the most of the Kanban method when applying it to end-to-end workflows. Such workflows usually span multiple departments and multiple stages in the lifecycle of some piece of work. Therefore, in the traditional enterprise they usually necessitate the collaboration of multiple teams. This is important to remember, as not all teams work on such end-to-end workflow, limiting the usefulness of the Kanban method and increasing the risk of the Kanban method not delivering to expectations (see this blog entry for more details: http://leanagileprojects.blogspot.se/2013/07/naked-kanban-dresses-in-teams_5.html*).*

The Kick-start is for the team: to help the team understand what it is doing, how, and in what context. Interestingly, the team's understanding that will surface during the Kick-start may not be aligned with the management's or the customers'. But that's OK! It is the best way to handle it, as the misalignment will be apparent once the team visualizes its work and policies. Once this it established and obvious, the next step is: what do we do to re-align?

Usually, the requirements on the Kick-start change with the level at which the team works. For example, teams that are at the top of a hierarchy of teams (e.g. program teams, portfolio teams, management teams, etc.), usually concentrate a lot of information that must be visualized in order to see and understand the whole and take right decisions. This may require several visualization boards: a board for exceptions management (issues, risks, blocks, opportunities, etc.), a board for status of deliverables (features, products, etc.) and a board for the team's actual work. This field guide does not (yet) gives support on these issues.

Team Manager

For the purpose of this guide, the team manager is responsible for the team to continuously improve: the manager creates and forms the team's environment to help the team to succeed. A Team manager can be a line manager, a project manager, a program manager, a unit/department manager and even a top/senior manager. Anyone that is responsible for managing and leading a team of individuals. The team manager is key to establish the improvement capability of a team, and one cannot succeed in creating lasting improvements without the managers' involvement. Also, a kanban system will make problems visible, and a team will lose its will to improve if the manager cannot, or will not, handle these problems.

Flow Manager

The flow manager is the person taking leadership of the kanban implementation within the team. The flow manager's role does not actually exist in the Kanban method (there are no prescribed roles whatsoever), but for the implementation to stick, there is still the need for someone – other than the coach – to take charge. Therefore, the purpose of the flow manager is to makes sure that the team follows its policies, creates new ones when needed, discusses and act on exceptions (issues and opportunities), experiments to find creative solutions, etc. The flow manager inspires, challenges and coaches. The flow manager really is an extension of the coach and is meant to take over when the coach phases out.
The flow manager can be anyone in the team (or even outside of the team). For development teams the team-lead usually, but not necessary, takes the role of the flow manager. For teams dealing with coordination and management it is common to have the team manager fulfilling the role.

Coach

The coach is responsible for helping the manager putting in place the necessary elements for the team to improve. For the purpose of this guide, this means preparing, running and following the Kick-start workshop. The coach is meant to be involved with the team for a while, until the flow manager takes over. Once the flow manager takes the lead, the coach spends his/her time coaching the flow manager and not the team directly. This allows the coach to be able to coach many teams, effectively scaling to a large organization.
In this guide, we assume that the reader is the Coach, i.e. the one setting-up, facilitating and following-up the Kick-start. The coach role may be fulfilled by anyone comfortable enough with the Kanban method, so that in real life the coach may very well also be the team manager or the team lead.
Within Sandvik IT, the Operational Excellence Support office has introduced the Lean Coach role that is fulfilled by experienced flow managers that want to go further.

Features

	Feature	Impact	Can it be skipped?
PREP 1	**Present Kanban as a Tool for Evolution**	Outcome: The manager (and flow manager) understands what the kanban method is and how it can be used for evolutionary changes. Output: a manager inspired use the Kanban method to pursue incremental, evolutionary change with the team.	Only if the manager and the flow manager are Kanban experts. *Run this feature even if the manager tells you that he/she already know about Kanban: more often than not people (managers especially) tend to think they master a subject after 5 minutes of googling around.*
PREP 2	**Study the Context**	Outcome: You understand the team's current condition and why it needs to improve. Output: A team status sheet.	Do not skip! *You cannot help the team without understanding the team, its context and its motivations to improve.*
PREP 3	**Set Expectations**	Outcome: The team manager, the flow manager and the coach have a common view on 1 What is expected from each of them for the change to succeed, 2 What needs to happen before the Kick-start, 3 What is the goal and expected result of the Kick-start, and 4 What happens after the Kick-start. Output: An OK from all parties in order to proceed with the Kick-start.	Do not skip! *There is a huge risk for the change not to stick if management is not committed and informed.*
PREP 4	**Prepare the Flow Manager**	Outcome: The person acting as the flow manager for the team understands his/her role. Output: A motivated flow manager	If the person acting as flow manager for the team already has a solid Kanban experience.
PREP 5	**Get a GO/NO GO Decision**	Outcome: The team manager takes responsibility for the implementation or backs-out. Output: A GO or NO GO decision from the team manager.	Always get a commitment from the team manager before starting any transformation.

These features may need to be complemented by other team-related activities in order to get the team ready for a Kanban Kick-start. These team activities are not covered in this field-guide. For example: Team members in a new team may need *to get together* and discuss how they want to work together (create a team charter, setup principles of conducts, etc.). This may also be an excellent activity for long-running teams in need of a fresh start.

PREP 1: Present Kanban as a Tool for Evolution

	Impact	Can it be skipped?
PREP 1	Outcome: The manager (and flow manager) understands what the kanban method is and how it can be used for evolutionary changes. Output: a manager inspired use the Kanban method to pursue incremental, evolutionary change with the team.	Only if the manager and the flow manager are Kanban experts and the decision to use the Kanban method has already be taken (so, usually no). *Run this feature even if the manager tells you that he/she already know about Kanban: more often than not people (managers especially) tend to think they master a subject after 5 minutes of googling around.*

You present the Kanban method to the team manager (and the flow manager).
A two hours meeting.

Supporting this presentation is out-of-scope for the field guide. Please check other resources for setting it up.

A good place to start is to mention the **foundational principles**, **core practices** and **agendas** reproduced here for your convenience.

4 Principles

- Start With What You Do Now
- Agree to Pursue Incremental, Evolutionary Change
- Respect the Current Process, Roles, Responsibilities and Titles
- Encourage Acts of Leadership at All Levels of the Organization

6 Practices

- Visualize
- Limit WIP
- Manage Flow
- Make Policies Explicit
- Develop Feedback Mechanisms at the workflow, inter-workflow and organizational levels
- Improve collaboratively using model-driven experiments

3 Agendas

- Sustainability
- Service-Orientation
- Survivability

PREP 2: Study the Context

	Impact	Can it be skipped?
PREP 2	Outcome: You understand the team's current condition and why it needs to improve. Output: A team status sheet.	Do not skip! *You cannot help the team without understanding the team, its context and its motivations to improve*.

You meet the team manager and the flow manager to understand the team and its context.
A one to two hours meeting is usually sufficient.

Understand the Purpose

Ask the manager: **What is the purpose of the team?**
There has to be a reason for why someone gathered a bunch of individuals to form a team; a goal that is bigger than what each individual can achieve alone.
Without a clear purpose, any change will be short-term.

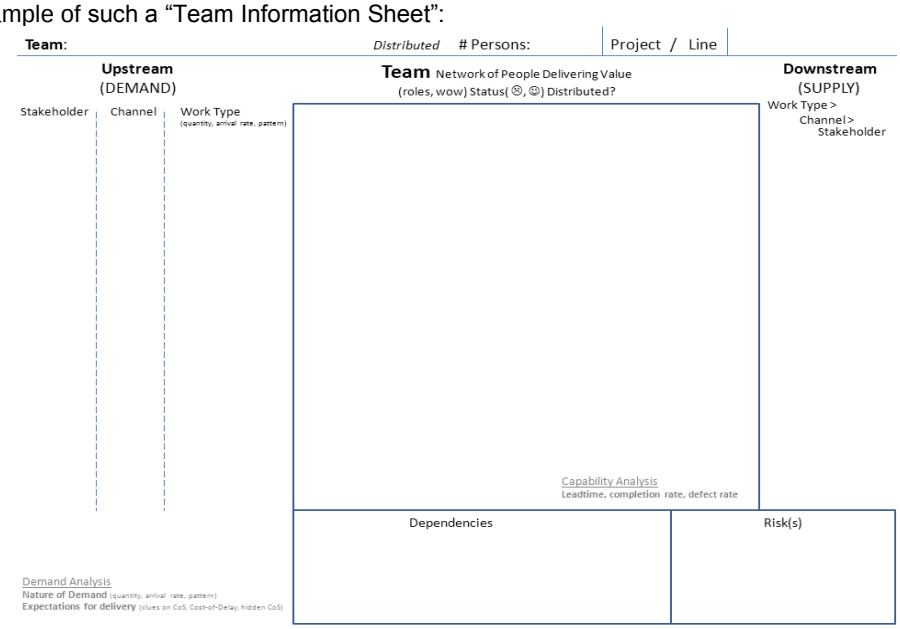
Understand the Context

Gather Data
Gather as much information as possible concerning **the team's current situation**:

- *How many persons are in the team?*
- *What are the roles of the persons in the team?*
- *What is the value stream (or parts of a value stream) that the team is responsible for?*
- *Who are the upstream and downstream partners for the team?*
- *What other teams this team is depending on?*
- *What other teams are depending on this team?*
- *Who is/are the customer(s)?*
- *How often does the customer(s) meet the team?*

If you find it practical, use an A3 paper-sheet during the meeting to draw the team, its dependencies, customers, etc. This will prove invaluable later-on as it will help you instantly remember this conversation when preparing the Kick-start and any following meetings and workshops.

Here is an example of such a "Team Information Sheet":



Analysis
Based on the gathered data, discuss with manager:
What are the current major risks for the team?
Record the risks on your information sheet.

Why Kanban?

Ask the manager:

- **What are your motivations for introducing Kanban?**
- **Why now?**
- **How do you see Kanban helping your team?**

Why did he/she approach you to introduce this "Kanban thing"?
What are his/her pains?
Record the motivations on your information sheet.

Acceptance Criteria

Ask the manager:
What are your acceptance criteria for this Kanban implementation?
In other words: If we project ourselves into the future, how do you expect the team to behave for this change (the introduction of Kanban) to be a success? How far in the future does this need to be?

Record the acceptance criteria on your information sheet.

Analysis

At the end of the meeting, or perhaps right after the meeting, ask yourself:
Given the context, risks, motivations and acceptance criteria from the manager, do I believe that I can help this team to improve using Kanban as an evolutionary change method?
Will your time be well spent, or will the whole story be a waste of time?

If you don't believe you can help the team, or if you have doubts, tell the manager that you need time to reflect. Go home and think about it once more. If you still are not convinced, politely refuse and give your feedback to the manager.

Some teams are in such a state that Kanban cannot help. For instance: teams without a manager or leader, teams in the middle of a major re-organization, teams that are not a team (individuals without a common goal), teams without purpose or customers, teams without work/demand, teams with serious conflicts, etc.

Extra step: Team retrospective

Discussing with the manager or team-lead you can sometimes get signals that the team is currently weak: lack of trust, internal conflicts, passive-aggressive behaviors, unclear internal roles, etc.

Do not go ahead if you feel unsecured about the team! Going ahead with the team in this state will be a waste of yours, and the team's, time. You need a focused team to succeed with the changes needed by the Kanban method. Sometimes such changes is all what is needed for a team to re-build itself, but do not count on it.

Instead, discuss the issue with the manager and decide whether a full retrospective could set things right again. If the manager accepts, and you believe you can pull it off, build a retrospective to clear the air and re-align everyone to focus on continuous improvements. Check the excellent "Agile Retrospectives: Making Good Teams Great" Esther Derby, Diana Larsen (2006) for tips on how to conduct such a retrospective.

If the retrospective is unsuccessful, give your feedback to the team manager and hope that the manager will have the courage and wisdom to fix things with time. Perhaps the team will be more focused in half a year or so…

PREP 3: Set Expectations

	Impact	Can it be skipped?
PREP 3	Outcome: The team manager, the flow manager and the coach understand: ● The purpose of the Kick-start. ● The kick-starting process. ● The prerequisites to the Kick-start. ● The Definition of Done. Output: An OK from all parties in order to proceed with the Kick-start.	Do not skip! *There is a huge risk for the change not to stick if management is not committed and informed.*

You set expectations with the team manager and flow manager.

Purpose and Process

You explain the purpose of the Kick-start:
- To introduce an evolutionary capability within the team.

You explain the Kick-starting process:
- What happens before the Kick-start workshop (understand the team, plan the workshop)
- What happens during the Kick-start workshop (run the workshop)
- What happens after the Kick-start workshop (reach a continuous improvement baseline)

Roles and responsibilities

You discuss the responsibilities for the different roles during the kick starting process.

Team Manager
The team manager is responsible for giving the team the support it needs to evolve. At the very least, the manager must be in regular contact with the team and address the issues the team will escalate.

Flow Manager
One of the key responsibilities for the flow manager is to challenge and question the team's policies. If a specific policy isn't used, why isn't it used? Should it be changed? And if so, why and to what?
More information on the flow manager's role can be found in PREP 4.

The Coach
The coach is responsible for kick-starting process. But, eventually, the coach should phase-out and let the flow manager take over. Therefore, the coach should work to train and coach the flow manager.

Prerequisites for The Team Manager

You check with the manager if the prerequisites for succeeding with a Kanban implementation are in place:

Do the team members have the authority and responsibility to decide on their way of working?
The manager should give enough trust to the team so that the team can decide how to best organize itself to do the work. This requires the team to constantly question the current standards and to seek smarter ways of doing the work.

Can the team decide when and how to PULL work?
One of the core properties of Kanban is to limit work in progress. During the workshop there will be a discussion on when to pull more work and who can decide that. Make sure you understand the Team managers view on that and if it is OK to give the team the opportunity to take responsibility for making that decision.

Has the manager time to be present?
The manager should participate in the team's planning meetings at least once a week. This is crucial for the team as it sends a signal that whatever the team is working-on is important to management (management is committed), and it allows management to get first-hand information on the team's current challenges.

Is there a clear policy on how to escalate issues that are outside of the team's control?
During the Kick-start, the team will learn how to flag problems that are outside of its control. Leaving these problems unaddressed is the best way to de-motivate a team. The manager must support the team in solving these problems.

Is there a policy for working with opportunities for improvements? What can team decide on its own? What requires approval from leader, manager, customer etc.?
A team can really become frustrated and lose motivation when it is not able to work on the improvements highlighted by a kanban system. It is therefore important to have a clear understanding - on beforehand - about how much time is available for improvements, what improvements are OK or not OK, what changes can the team make on its own, what changes require approval, etc.

Is it SAFE for the team to fail?
Improvements that make a difference can only be found by a series of trials and errors. Team members should feel safe to try small experiments that, sometimes, will fail.

Prerequisites for The Flow Manager

You check with the flow manager if the prerequisites for succeeding with a Kanban implementation are in place:

Who will be the Flow Manager?
The flow manager in a Kanban Kick-start is in charge of facilitating the introduction and the sustained usage of the techniques and methods presenting during the kick-start. Eventually, the flow manager is responsible for the continuous improvement of the team.

Does the Flow Manager have time?
For a good start, the flow manager should have about 25% of his/her time to spend on team: that is making sure that policies are followed, blocking issues are removed and the team improves.

Does the Flow Manager have the required dedication?
It is often hard for a team to see and understand what needs improving. It is therefore important that the flow manager is truly dedicated to tirelessly visualizing the team's current situation, showing what is going-on and asking the right questions; so that the team becomes aware of the current problems and decides to act on them.

Prerequisites for The Coach

Do you have time to follow-up the team after the Kick-start workshop?
You will need time to at least attend some of the team's planning meeting and to facilitate a couple of follow-up meetings.

The Definition of Done

As a coach when are you done with the team?
If you are the flow manager or manager for the team, then you are only done when your engagement with the team ends.
If you are a coach taking care of several teams, take time now to discuss and agree with the manager how long you will support the team.
Options are:

- *Just Kick-start.* You kick-start the team and let it navigate alone the raging seas.
- *Bring the team to continuous improvement baseline*. You support the team until it reaches a baseline (this may take months for some teams).
- *Bring the team to continuous improvement baseline and set target conditions.* You help the team evolve to become what it must.
- *Forever.* And ever.
- *X weeks/months*. Regardless on the state of the team by then.

PREP 4: Prepare the Flow Manager

	Impact	Can it be skipped?
PREP 4	Outcome: The person acting as the team-lead for the team understands his/her role. Output: A motivated flow manager	If the person acting as flow manager for the team already has a solid Kanban experience.

You prepare the flow manager to be ready for the Kick-start.

Does experience matter?

An experienced flow manager will help a team to mature and improve much faster. An inexperienced flow manager will mature with the team and may need more time to get the team to an 'evolve' state.

A flow manager with experience in Kanban doesn't really need an introduction. Though, as the flow manager will be your main contact-person with the team, you may still want to discuss the role and how you will interact with each other.

About the Team-Lead's Role in the Kick-starting Process

Discuss the need for a flow manager in the Kick-starting process.

(The following is just some ideas about what to discuss)
The need for a flow manager:

- To facilitate, coach and challenge the team
- To escalate risks/issues/problems/opportunities
- To communicate and "strike a bargain" with up- & down-stream partners

Principles

- *Facilitate* (introduce, sustain and improve the method)
- *Coach* (learn the team to see & understand waste / problems / risks / opportunities)
- *Challenge* (place the team "outside" of its comfort-zone)

The Role of the Flow Manager over time

Discuss how the flow manager's role will change over time.

(The following is just some ideas about what to discuss)
Before the Kick-start

- Advertise for the need of a change (Get in control? Be better? Become excellent?).
- Get everyone interested/on-board.

Shortly after the Kick-start

- Make sure the team follows the method (meetings, visualization, etc.).

When the team starts to "get it" (any time after the kick-start)

- Make sure that the team members consider the Kanban board theirs (e.g. team members take turn to facilitate the board).
- Teach the team members to see & understand waste, blocks, and bottlenecks and have the team act on it.
- Strike a bargain (manage customers/partners).

Improve Continuously

- Always keep the team "on the edge" by challenging the team members to improve (do not let the team linger too long in its "comfort-zone").

Responsibilities

Discuss the responsibilities of the flow manager.

(The following is just some ideas about what to discuss)
The flow manager has the responsibility to make sure that:

- **The team visualizes** its current *work-in-progress*, *work process*, *policies* and *improvements* using a *visualization board*.
- With the help of the visualization board, the **team meets regularly** to plan **the smartest way** to bring the ongoing work to completion.
- All team members have **commonly agreed** on the team's policies.
- The team **follows all commonly agreed policies**, daily.
- The team takes time (after a "planning meeting" or during a *Retrospective*) to **add, discuss and change any policy** that is not used, became obsolete or wrong.
- The team meets regularly - at least once per month - to **reflect and learn** on its current way of working. This can be done in a separated meeting (*Retrospective* or *Kaizen event*) or as an extension of the planning meeting. The teams' current policies are then updated to reflect the result of the retrospective.
- All ongoing customer-related has been **approved by the customer** or his/her representant.
- Team members **focus on finishing the ongoing work** before starting new work (this may require the team members to help each other).
- The team uses a **Cumulative Flow Diagram** (CFD) to see how the work-in-progress and average lead-time is evolving over time. The CFD is updated at least once per week.
- The team knows the current **lead-time for each Work Type** defined by the team.
- The team **logs all changes** done to the current way-of-working (policy changes, DoD changes, visualization changes, etc.).

The purpose of this list is to kick-start flow managers that are new to Kanban into action. It states the basic minimal requirements that one can expect from a flow manager. It is by no means an extensive list of all the responsibilities. As the flow manager matures, there are many more - or less - things to do, investigate and improve.

Recommendations

Give the following recommendations:

- Get in touch with the coach as soon as there is a problem/question
- Flag the coach as soon as possible when:
    - There is no management attention.
    - The team's energy is spent in conflicts (within the team or with management or customer).

PREP 5: Get a GO/NO GO Decision

	Impact	Can it be skipped?
PREP 5	Outcome: The team manager takes responsibility for the implementation or backs-out. Output: A GO or NO GO decision from the team manager.	Always get a commitment from the team manager before starting any transformation.

You get a GO/NO GO decision from the Team manager to proceed with the Kanban implementation

Should I stay or should I go?

If you believe that you can help the team to improve using the Kanban method (based on the result of PREP 2), ask the team manager: **Now that you understand the benefits of the Kanban method and what it requires to succeed (incremental evolutionary change, leadership, flow manager), do you agree to proceed with the Kanban implementation?**

If it is a *GO*, the only thing left to do is to agree on a **date** for the Kick-start.

If it is a *No GO*, make sure you understand the reasons behind the decision. Even if you experience it as backlash, there is some very interesting information to unearth here about your company's inability to improve.
Do you agree with the decision? If yes, what are the main impediments? If no, what would you have done differently during the preparation phase to get to a GO? Can you contribute to lift some of the impediments that contributed to the decision? What organization issues are in the way? What should you escalate for resolution?

In any case: *Good luck!*

Before the Kick-start: Plan the Workshop

You understand the team's context and the expectations are clear for all involved. Now is the time to prepare the actual workshop.

Definition of done:

- *You have an agenda for the workshop that fits the team's current situation.*
- *You have booked the time and place for the workshop and invited the whole team.*
- *You have addressed various practical details for the workshop*

Features

	Feature	Impact	Can it be skipped?
SETUP 1	**Create an Agenda**	Outcome: You have reflected on how to best meet the team's need during a workshop. Output: An agenda for the Kick-start that fits the team's needs.	*If you run the same standard Kick-start setup for all your teams and you already have an Agenda.*
SETUP 2	**Setup the Workshop**	Outcome: All workshop attendants are notified about where and when the workshop will be held. Output: You have booked a suitable room, you have invited the right persons and you have the right materials to run the workshop.	Never!
SETUP 3	**Extra Preparations for a Distributed Team**	Outcome: If the team is distributed, you know how to handle the day-to-day communication. Output: the team has the right tools for communicating.	Skip if the team is collocated.

SETUP 1: Create an Agenda

	Impact	Can it be skipped?
SETUP 1	Outcome: You have reflected on how to best meet the team's need during a workshop. Output: An agenda for the Kick-start that fits the team's needs.	If you run the same standard Kick-start setup for all your teams and you already have such a setup.

You create an agenda for the Kick-start that fits the team's current situation and maturity.

Select Features

Based on your understanding of the team, you now need to create an agenda for the Kick-start workshop. This is essentially answering the question: *What does the team needs to go through during the workshop for it to be able to start using Kanban the very next day?*

The "Run the Kick-start Workshop" section lists a collection of "features" that can be used to build a Kick-start workshop. Use this list to select the features that you think the team needs to go through during the Kick-start. The default Kick-start workshop is to use all features in the given order.

Go through the list and select the features you think are needed for the team based on the "when can it be skipped?" criteria.

The Kick-start workshop must fulfill three goals:

1. The team members must get an understanding about the team's purpose, context and way of working.
2. The team members discuss and agree on how to visualize the team's current work, workflow and policies.
3. The team members know how to use the visualization board to take smart decisions.

In what order should the features be completed?
The list of features under the "Run the Kick-start Workshop" section suggests a sequence in which the features should be completed. You are encouraged to change the sequence as you see fit. For example, we have found that starting with a mini retrospective helps the team members to get involved in the workshop early, though you may prefer to have the retrospective after you have explained the Kanban mechanics or after discovering what the team does.

Pauses

Plan to have regular pauses to keep everyone fresh (a 10 min pause every hour usually works well). It is a huge plus if you have the possibility to take your breaks somewhere else than in the workshop room.
Plan to have plenty of coffee, water, fruits, pastries, etc. available during the pauses.

When announcing a pause, always summarize what the team did and achieved since the last pause and what is the team should expect after the pause.

SETUP 2: Setup the Workshop

	Impact	Can it be skipped?
SETUP 2	Outcome: All workshop attendants are notified about where and when the workshop will be held. Output: You have book a suitable room, you have invited the right persons and you have the right materials to run the workshop.	Never!

You go through all the little practical details for holding the workshop.

Who should attend?

All team members should be present **physically**. If your team is not collocated, the workshop is the perfect occasion to bring everyone together and plan some team activity.

The team manager **must** be present during the whole workshop. Do not run the workshop if the manager has no time or interest to be present: this will become **a major** impediment that will work against all you try to achieve with this workshop.

Kick-starting several teams at the same time?
A Kanban Kick-start as described in this guide tries to fit a lot within one day and requires a lot from the facilitator. For this reason, Kick-starting several teams at the same time has not worked well for us, even with several facilitators. Each team will need to spend more time on different features, making the synchronization of the workshop very difficult. In the end none of the team will have a great workshop. So, avoid!

The Team's Board

If you are using an "analog" visualization board for the team try to get a "white-board-on-wheels". This will make it easy to bring the boards to different meeting rooms for follow-ups, retrospectives, meetings with the customers, etc. Just make sure that the board is small enough to be able to navigate in your office (Doors! Elevators!).

Props

To create an "analog" board, the team will need:

- Stickies of at least 5 different colors.
    - Try to get select very contrasting colors to accommodate for color-blind people.
    - Invest in "real" stickies or "extra-sticky" stickies: next morning you do not want to find all the stickies as a random pile on the floor!
- White magnets (at least 3 per team member).
- Some red magnets.
- One "fat" pen per person (sufficiently "fat" so that what may be written on a sticky can easily be seen by all in the room).

The Workshop Room

Space

- The room for the workshop should be large enough for everyone to have a place to sit as well as enough room to stand in front of a board.
- If you are using an “analog” board for the team, you should have enough room for the board.
- Use a big plastic sheet or brown paper as a temporary board if you do not have enough room or if you cannot move the “official” team’s board. The team will have to “copy” the temporary board to the real board after the workshop.

Setup

- Have flip-charts ready to be used somewhere in the room.
- If you have a group of more than 10 persons, set up the room to create small islands where 3-6 persons can sit together.
- Do not allow laptops during the workshop, except:
    - If someone is responsible for handling incidents and such.
    - When team members need information to create work items when populating the board.

Invitation

Send an invitation to the workshop to all attendants.

What should the workshop be called?
Actually, try avoiding calling the workshop a “Kanban Kick-start”. This puts the focus of the workshop on a method/tool/solution. At best, team members will come to the workshop in a passive state of mind as they expect to be served a ready-made solution. At worst, some of the team members will have read all about Kanban and may have the wrong focus and expectations during the workshop, even lock themselves in some erroneous interpretation.
Instead, call the workshop an “Improvement Kick-start” and the team members will expect to do most of the work by themselves, really having the right mindset for the day.

SETUP 3: Extra Preparations for a Distributed Team

	Impact	Can it be skipped?
SETUP 3	Outcome: If the team is distributed, you know how to handle the day-to-day communication. Output: the team has the right tools for communicating.	Skip if the team is collocated.

You have discussed and prepared the tools needed for the team to communicate on a day-to-day basis when working distributed.

Collaboration Tools

Discuss with the team manager how (by which means) the team members will collaborated on a day-to-day basis.

The solution should at least cover:

- A way to have the visual board visible to all team members.
    - Take pictures of the board when needed
    - Use video to share the current board
    - Use a digital board solution (there are many products available, Sandvik IT recommends Lean Kit Kanban)
- A way to conduct daily planning meetings adapted to the fact that all team members cannot be physically present.
    - Telephone conference
    - Video conference
- A way to discuss and resolve issues by engaging in a conversation between two or more persons not physically able to meet.
    - Telephone conference
    - Video conference

Run the Kick-start Workshop

You have booked the workshop, addressed all the practical details, and you have an agenda that fits the team you are going to Kick-start. Now is the time to run the one-day workshop.

Definition of Done:

- *The team has a board visualizing the on-going work, the most important work that is ready to be started, the workflow (or way-of-working) for the team, and various policies governing the work.*
- *The team has discussed and agreed on various visualization and work-related policies. These policies describe the team's current way-of-working (no "wanted state" here).*
- *The team has a way to work with the board to plan the team's activities.*

Features

	Feature	Impact	Can it be skipped?
KICK 1	**Set the scene**	Outcome: To answer the question "Why a workshop?" and set the goals for the workshop. Output: a formulated *Definition of Done* for the workshop, an agenda and other practical details.	Never! Always start the workshop by setting the scene.
KICK 2	**Share Current Concerns**	Outcome: To focus the team on their current situation and to "open-up" team members to discussion. Output: Identified the issues that can be addressed during the workshop and those who can't.	If you have recently run a retrospective with the team.
KICK 3	**Define a Shared Vision**	Outcome: To have the team discuss and agree on its purpose. (Without a clear and shared purpose there cannot be long lasting improvements). Output: a slogan describing the team's purpose, some criteria for the team's success.	If the team members have a clear and shared purpose.
KICK 4	**Discover what the Team Does**	Outcome: To really **understand** the team's current situation (what the team actually does as opposed to what the team thinks it does). Output: a model of the team with an agreement on when/where the team's responsibility starts and ends.	If the team has already a good understanding about is situation/context.
KICK 5	**Identify Work Types**	Outcome: To have the team understand what type of work it usually handles. Output: a list of work types	If the team already has valid work types, or if you want to force the team to use pre-defined work types (be sure that they make sense in the team's context).

KICK 6	**Explain how the Board works**	Outcome: For the team members to understand the mechanics of a kanban system (Ready/A/B/C/Done, Pull, Blocks, etc.)	If all team members are already familiar with Kanban. (It may still be a good idea to go through that step to introduce/refresh some concepts like CoS or Capacity Allocations).
KICK 7	**Present a Common Language**	Outcome: To introduce a language/vocabulary (to use when building a board) that is common to a whole unit/ department/ company. As a result, different teams use the same words for the same things (e.g. everyone knows what the “Prepare” stage means in a workflow).	If you do not have many teams to harmonize or are missing a common vocabulary.
KICK 8	**Set Visualization Policies**	Outcome: To agree how on to visualize work in order to understand what is going on and take appropriate smart decisions. Output: a list of policies governing the visualization of the workflow (so that they can be remembered and shared with others).	If the team already has visualization policies that do not need to be revised.
KICK 9	**Create & Populate the Board**	Outcome: To visualize the work currently handled by the team, according to the existing policies. Output: a physical board full of colorful stickies, or a digital board full of work items.	If the team already has a board that visualize all on-going work items according to the current policies.
KICK 10	**Set Way-of-Working Policies**	Outcome: To agree on how to handle work. So that all team members handle work in a consistent way. Output: a list of policies governing the way work is handled.	If the team already has way-of-working policies that do not need to be revised.
KICK 11	**Set Limit WIP Policies**	Outcome: To agree on how much work is currently optimal for the team. Output: a policy dictating how much work is allowed on the board (per column, swimlane or person).	If the team already has a satisfactory WIP limit policy, OR the team isn’t mature enough yet to sustain the policy.
KICK 12	**Set Planning Meetings Policies**	Outcome: To agree on how to use the board to take smart decisions. Output: a policy dictating when to meet, how (especially for distributed teams), who holds the meeting, the purpose of the meeting and its details.	If the current meeting policy exists and is satisfactory.

KICK 13	**Run the first Planning Meeting**	Outcome: All team members understand how the meeting works. Output: the first planning meeting is done.	If all team members are well used to planning meetings.
KICK 14	**Close the scene**	Outcome: the team is now in charge! Output: feedback about the workshop for you to improve.	Never! Always end the workshop by closing the scene, except if you *really* have no time left (here is an improvement for you: keep time for closing the scene).

KICK 1: Set the Scene

	Impact	Can it be skipped?
KICK 1	Outcome: To answer the question "Why a workshop?" and set the goals for the workshop. Output: a formulated *Definition of Done* for the workshop, an agenda and other practical details.	Never skip this step!

You set the scene for the workshop: Why? The Goal. What will happen?

Welcome!

- Introduce yourself, your role and responsibility (if necessary)
- Explain your role today during the workshop (facilitator) and why you are facilitating.

Always start with WHY

- Let the team manager explain the reasons (issues) **why** this full day workshop was necessary.
- Briefly describe how these issues will be addressed during this workshop.

Definition of Done

At the end of the day the team will have a **visualization board** (sticky-powered or digital).
The board will be used:

- To **visualize** the team's ongoing **work.**
- To **visualize** the team's current **workflow** (avoid using the work "process" as it is quite overloaded).
- To **visualize** the team's current **policies** (socially enforced principles that govern how the work works).
- As a result:
    - the team will have enough information to see and understand it current condition/situation,
    - So that the team will be able to improve on its situation.

Key Concepts

- The visualization is **by the team and for the team** (not for a manager, a flow manager or the customer).
- **We will not introduce a new way of working!** We will only visualize and make explicit how the team is **currently** working.
- Actually, we may introduce **one** feature that may impact to the current way of working: *team members will **pull** work when they have capacity for it*. This means that the team may get more responsibility than it currently has (make sure that you have discussed this with the team manager and/or coordinator prior to the workshop).
  Actually this Pull feature is not strictly necessary for a kanban system, but it greatly helps the transition from a stressed situation to a more stable and sustainable situation for the team members.

Kick-start?

Kick-start means that *the team is in charge*, not the facilitator!
During the workshop, and with the help of the facilitator, the team will set in place the elements needed to start improving, but the effort needed to identify the improvements and to realize them is up to the team.
After the workshop, the facilitator/coach will regularly follow the team to make sure that the elements are setup and used properly, but it is the team's responsibility to use them.

The Golden Rule

Keep it Simple and Good Enough!

- The workshop is time-boxed: there is no time to find the *optimal* solution for everything! Some issues will need the team to mature before they can be resolved satisfactorily (and this may take months).
- Therefore, decide *something* (sometimes “anything” will do) and see it as an experiment. In the coming days or weeks you will see how well this assumption holds when confronted with reality. This will give you the necessary feedback to be able to try another assumption that, hopefully, will work better in your context.

Present the Agenda

- *Safety First* (Sandvik meme): indicate where the nearest exits are located in case of an emergency.
- **Present the features** that you have selected for the team for this workshop.
- Explain that there will be **frequent pauses**, and explain what to expect during the pauses (where to go, coffee, fruits, etc.).
- Have the team agree on a **time for lunch** (if the time is negotiable).

Visualize the agenda as a mini Kanban board. One sticky per item. All items start in the “ready” queue, then during the workshop, just pull each item into “doing” and then “done”.

Last thing: Ask for someone to act as a **secretary**! This person will write down policies for the team as they are discussed by the team (the facilitator has better things to do).

Now you are ready.
Let’s do it!

KICK 2: Share Current Concerns

	Impact	Can it be skipped?
KICK 2	Outcome: To focus the team on their current situation and to "open-up" team members to discussion. Output: Identified the issues that can be addressed during the workshop and those that can't.	If you have recently run a retrospective with the team.

You facilitate a quick retrospective to help the team members share their current concerns and identify those that may be resolved during the workshop. The real goal for this feature is to engage the team members and place them in a problem-resolving mode.

Gather Input

Ask each person to identify his/her Top 3 issues and write them down individually on a sticky.
Increase the limit to 5 stickies if the team is small. Time box this step to 5 minutes.

The issues should be *problems, irritations, missing things, things that are in the way*. Anything that makes the team not being *effective* or *efficient*, that makes the work *stressful* and/or *dull*, that makes the work not *flow*.

Share Input & Group Similar Issues

Ask each person to go to the board and quickly explain his/her issues.

- Ask each person to place the stickies on the board.
- And explain the meaning of the sticky to the rest of the team (and not to the coach!).
- When everyone has spoken, quickly group similar stickies into groups.

Sort

You sort the groups of issues into 3 distinct categories:

- **Way-of-working or Policies issues:** issues that will be discussed (and maybe resolved) during the workshop, as they relate to *how* the work is done, how the team *behaves* or how other groups *interact* with the team. *Place these issues on a flip-chart for all to see and refer to during the workshop*.
- **Other issues** that cannot be discussed and/or resolved during the workshop.
  Place these issues on a separated flip-chart and ceremoniously give the chart to the team manager to be addressed after the workshop.

Big Team (>10p)?

Divide the team into smaller groups (3-6 persons) and ask each group to identify issues. Increase the number of issue from 3 to 5 per group.
Make sure that the groups are balanced (number of persons and "types" of persons)!

KICK 3: Define a Shared Vision

	Impact	Can it be skipped?
KICK 3	Outcome: To have the team discuss and agree on its purpose. Output: a slogan describing the team's purpose, some criteria for the team's success.	If the team members have a clear and shared purpose.

You help the team understand its purpose.
Without a clear and shared purpose there cannot be long lasting improvements.

The Purpose

In order to help the team members to reflect on the team's purpose, ask the following (quite provocative) questions:
"Can someone tell me Why do you exist as a team?"
"Why would someone pay hard earned money to keep you busy?"
"What is the value of what you do?"

This certainly fires-up discussions and needs some facilitation to keep the discussion straight and meaningful. **Write down all the keywords** you hear the team members mention, group them, refine them, validate them and acknowledge them.

Answers like "because the customer wants it" are not sufficient. In this case, the team members must place themselves in the customer's or stakeholder's shoes and try to find a higher value to their work.

Rather quickly, some sort of meaningful purpose should surface. The goal of this exercise is not to find the right/best purpose for the team, but to get the team members to agree on something. Regardless of how "good" the result is, this agreed purpose will serve for the time being. Any discrepancies from the "right" one, or "planned"/"envisioned" one, is very interesting to investigate - though at a later stage, as it is not the purpose of the workshop.

Create a Slogan

Ask the team: *"**Can someone give me a 'one-liner' or 'slogan' that captures your purpose?"***

Give the team 5 minutes to come up with something that most team members can agree upon and write the slogan on a flip-chart (it can be written on the team's board when creating the board). Again, this slogan is not meant to be right for ever, it should capture the current purpose of the team.

If nothing comes up after 5 minutes, skip the slogan but write down the most relevant keywords that the team agreed upon.
Examples:

- *A team working with bringing new systems into production "We act as translators between customers, operators and suppliers".*
- *A team working with SAP authorizations "We make sure that SAP is used in a way that benefits the organization".*
- *A team working with end-user experience "We manage end-users expectations".*

Troubleshooting...

If nothing meaningful is surfacing after 20 minutes of discussion: take a break! During the break, discuss with the manager how to proceed. This workshop is meant to set in place a mechanism for evolutionary change, without a valid purpose (shared and understood by all team members) there is no possible direction for any improvement. If the manager still want to proceed with the workshop: skip this step. Though, the first target condition to set with the manager after the workshop is to find a valid purpose for the team (this may require the involvement of upstream and downstream partners, as well as stakeholders and customers).

Criteria for Success

Take 5 minutes to ask the team "**Now that you have defined this purpose, what are your key success factors? What is crucial for you to succeed?"**
Write down on a flip-chart the key words that the team may throw at you. These keywords will be essential when creating the policies later on.

Here is an example: key success factors for a team that deals with integration of business applications:

- *Business knowledge (or at least a documentation of such a business knowledge)*
- *Or, access to business specialists (that have time for us)*
- *Technical skills in integration platforms*
- *Be involved early in a project that contains integration elements*
- *High level of quality (integration incidents can have huge financial consequences)*
- *Have an end-to-end perspective (any integration may involve several platforms)*
- *Ways to identify work that will create new integration flows as it is how we get paid.*

Team composition

Ask the team ***"Is everyone needed for achieving the team's purpose in the team?"***

If not, who is missing? Shouldn't that person be here with us today?

If the team members realize that they are missing someone: take a break, discuss the issue with the manager and decide how to proceed. If you continue, make note of who is missing and, when visualizing the workflow later on, ask the team how these persons are involved in the team's workflow (some may 'become' column on the board or may be required in some definition of done).

Big Team (>10p)?

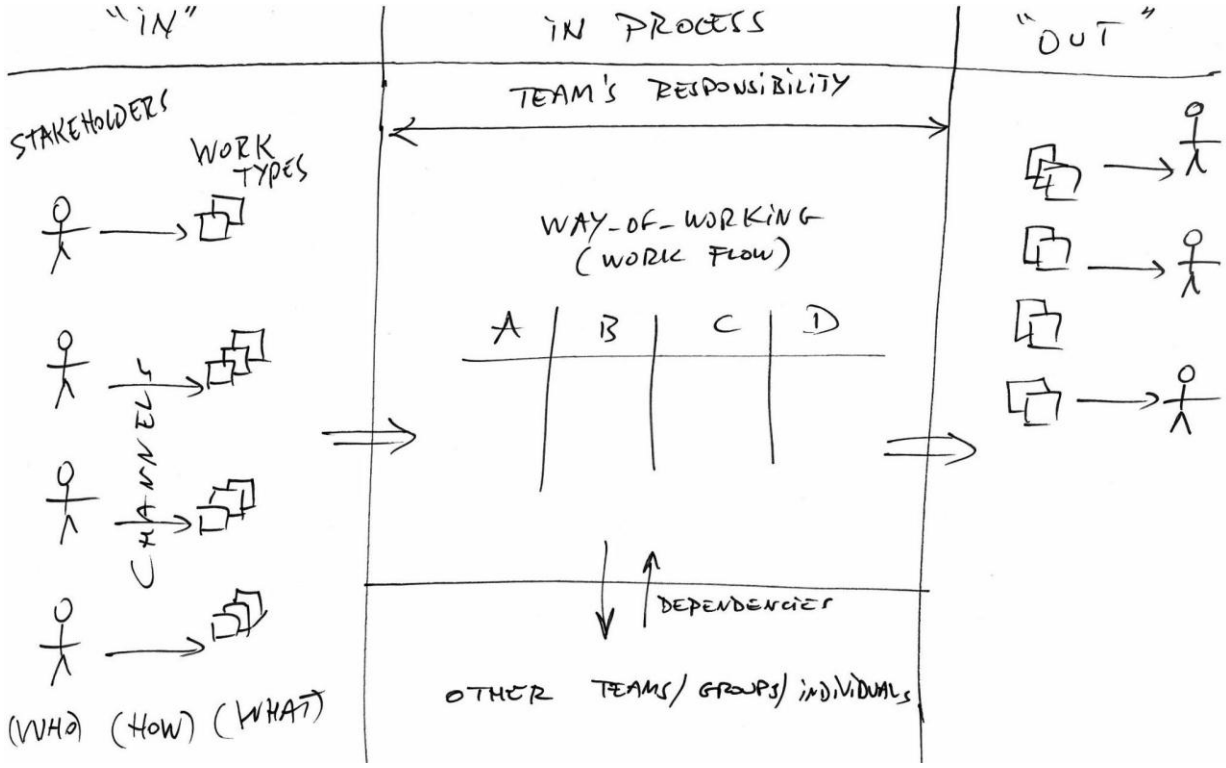
Divide the team into smaller groups (3-6 persons) and ask each group to identify a purpose and slogan. Then, have each group present its purpose and slogan. Facilitate a discussion between the groups to come up with a unified purpose and slogan for the team.
Re-use the same groups as earlier if possible.

KICK 4: Discover what the Team Does

	Impact	Can it be skipped?
KICK 4	Outcome: To really **understand** the team's current situation (what the team actually does as opposed to what the team thinks it does). Output: a model of the team with an agreement on when/where the team's responsibility starts and ends.	If the team has already a good understanding about is situation/context.

You help the team understand its context: who wants what from the team and how the team usually handles these requests.

During this step, we will build a model of the team by considering the team as a closed system and analyzing its interactions with the rest of the organization (a system view). The resulting model will allow the team to understand its responsibility and the complexity of its situation.
Here is a model of such a system view:



A System View

Upstream

Ask the team to identify all the *Stakeholders* that create demand for the team.
Anyone that want some work (demand) done by the team. The team may refer to them as stakeholders, customers, partners, managers, architects, other teams, etc.
The team itself may be a valid party here (e.g. improvement work).

Ask the team to list *examples* of typical demand, or work, which originate from each stakeholder.

For each work item, ask the team to list the *channels* the demand takes to reach the team.

Ask the team members to quantify how often and how much of a demand they see in (say) a week.

The goal here is for the team to understand the nature of the demand, who generates that demand and how does it reach the team. This information will be use later on to categorize the demand into Work Types.

Downstream

Ask the team members who (other teams, individuals, partners, etc.) is consuming the result of their work (if relevant).

Ask the team to list typical "completed" work items.

Ask the team to identify the channels the complete work takes to reach the consumers.

Ask the team how often they deliver finished work downstream.

The goal here is for the team to think in term of "consumers" or "downstream partners". This information will be used later on to categorize the demand into Work Types.
This will also prove valuable when seeking feedback from these consumers. The team may even discover issues during this discussion that may be addressed by a having a better Definition of Ready or Definition of Done (see later during populating the board).

Agree on when and where the responsibility of the team starts

Ask the team to discuss and agree on where and when the responsibility of the team starts.

- Is it when the demand is received?
- By who?
- When/where?
- Is it different for different demands?
- If so, can the team come up with a simple rule that works for all work types?

This information will be needed for setting way-of-working policies later on.

Agree on when and where the responsibility of the team ends

Ask the team to discuss and agree on when and where their responsibility ends.

This may be different for different completed work types and consumers.

The goal here is for the team to agree on the scope of their work. Different team members may have different opinions on that.

Within the Black Box

Within the responsibility of the team:
Identify the different steps, or phases, the demand is going through to be brought to completion.

Ask the team to explain how they usually handle a demand once it is under their responsibility. What is the workflow (or "way-of-working", or "process" - though this last one usually has negative connotations)?
To help the team during this step, make the team focus on its most common demand.

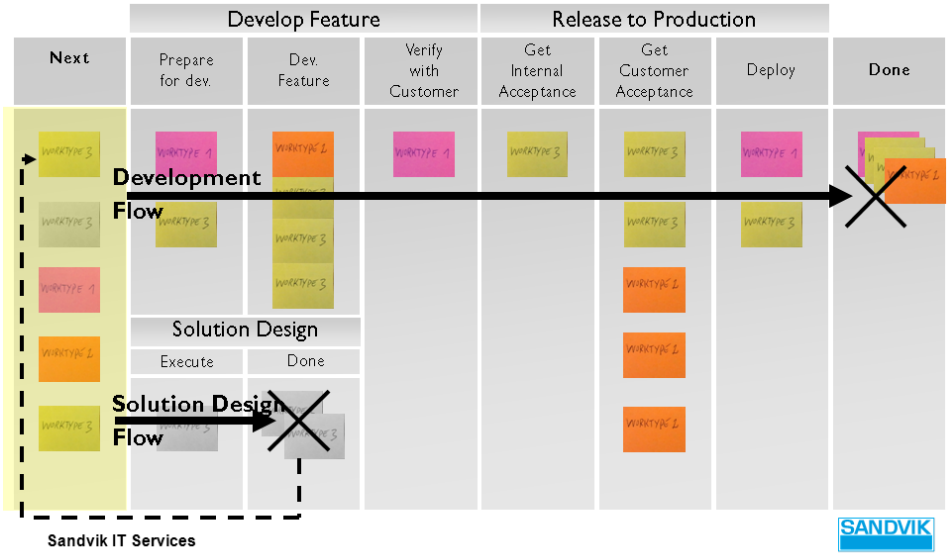
Ask the team to draw the workflow as stages following each other's.
Which stages, or steps, are the work items going through? What are they called?

Try to keep it simple!
The visualized workflow doesn't need to map 1-to-1 to the team's process (actually, it probably shouldn't).
The workflow may be too detailed if you have more than 10 stages. Later we will introduce the concept of Definition of Done (DoD) for a stage: it could be that some of the columns you have now will eventually disappear and become exit criteria in a DoD.

Try to keep it at a rather high level!

Each work type may be treated differently and technically require a separate workflow, but drawing and visualizing too many workflows might just be too confusing.

Can the team abstract the different workflows into one?
If this proves hard, it can be Ok to visualize two workflows: one specific to the team's domain where most of the work items are treated, and one very generic ("Doing" and "Done") for those work types that do not fit in the previous flow.
Here is an example:



Note that a portfolio, program or project may need some complementary views/boards in order to get the whole picture. Beyond a standard "work" board (for the work that the team members are "working" on), these teams at a higher tier/level may need to visualize: the work in progress at lower tiers (e.g. a program what to see what is happening for each project) as well as exceptions (issues, risks, blocks, opportunities).
We may provide more guidance on this in a future version of the field guide.

Identify dependencies to other groups while work is in progress

Ask the team to identify the dependencies it has on other groups, teams, initiatives or individuals while being responsible for the completion of a work item.

These dependencies can be visualized either by a dedicated column on the board with a queue in front of it, or by a using a block on a work item.

Big Team (>10p)?

Divide the team into smaller groups (3-6 persons) and ask each group to directly come up with different Work Types (max 5). Then, have each group present its Work Types. Place the Work Types on a board and categorize them according to whether they describe demand, internal activities or completed work. Facilitate a discussion between the groups to come up with max 5 demand Work Type. Use the "internal activities" and "completed work" work types to create a workflow.
Re-use the same groups as earlier if possible.

KICK 5: Identify Work Types

	Impact	Can it be skipped?
KICK 5	Outcome: To have the team understand if it is working on the right things (balance). Output: a list of work types	If the team already has valid work types, or if you want to force the team to use pre-defined work types (be sure that they make sense in the team's context).

You help the team categorize the demand into different types of work.
In practice, this step is often done in conjunction with KICK 4.

Gather Data

Based in the information gathered during KICK 4, **Ask the team to flag the work items that:**

- Have the most ***value*** for their customer(s) (e.g. mark them with a *).
- Usually have a ***higher urgency*** than the rest (e.g. mark them with a !).
- Generate the ***largest quantity*** of demand (e.g. under-strike them with a fat line).
- Generate the ***least quantity*** of demand (e.g. under-strike them with a dashed line).
- Are directly ***aligned with the team purpose*** (e.g. encircle them). These are the work items that the team would like to do more of.
- Are consider so ***remote from the team purpose*** that they can be considered waste (e.g. cross them). These are the work items that the team would like to do less of.

Analysis

Ask the team to discuss what they see.
Ask the team to group all these different work items into a maximum of 5 Work Types (more than 5 work types are hard to remember and visualize without overloading the board).

This step is really about the team members understanding the demand on the team. The work types the team comes-up with may not be ideal nor right (according to customer, top management, other teams), but it mirrors the current understanding the team has of its purpose and demand. Using these work types in real life will help the team members evolve their understanding, especially when discussing them with customers and management. Then, and only then, can the team members change the work types to better fit this new understanding. This is of course a continuous process. Work Types can be based on:

Source of the work
- System X
- Customer Y
- Improvement work
- Maintenance work.

Flow of the work
- Development
- Pre-study
- Requirement gathering

Size of the work
- XL/L/M/S/XS

Risk profile of the work
- Strategic initiative
- Regulatory work
- Standard work

Outcome of the work
- Release to production
- Report (pre-study)

Relevance of the work
- work we should do more of
- work we should do less of

Here are some examples of work types that some teams came-up with.
Maintenance Software team: Incidents, Changes, Pre-studies, Improvements, Others.
SharePoint team: Support, Standard-work, Custom-work, Pre-studies, Improvements, Consulting.
Program Coordination (coordinating several projects): Issues, Risks, Dark Matter, Opportunities, Activities.

KICK 6: Explain How the Board Works

	Impact	Can it be skipped?
KICK 6	Outcome: For the team members to understand the mechanics of a kanban system (Ready/A/B/C/Done, Pull, Blocks, etc.)	If all team members are already familiar with Kanban. (It may still be a good idea to go through that step to introduce/refresh some concepts like CoS or Capacity Allocations).

You present the mechanics of the Kanban board to the team.

Here are some of the concepts to cover:

Concept	Visualization	Outcome
Work Item	Sticky.	To see the work.
Work Item size (estimate)	A T-shirt size on the top right corner of the work item sticky. *XL > 3 weeks work for 1 person* *L 1 - 3 weeks of work* *M < 1 week of work* *S 1 - 3 days of work* *XS < 1 day of work*	To see how much work (and risks) is abstracted by the work item sticky. It may make sense to limit the size of work items (break them down). Of course, the size is just an estimate and may be wrong.
Work Type	Color of the sticky.	To see the current mix of work being managed. Is this the right mix?
Workflow	Columns on the board. Usually starting with 'Ready' and finishing with 'Done'.	To see the steps needed to bring a work to completion.
Pull		A mechanism to avoid choking the system with too much work in progress. It tries to limit the amount of work to the team's current capacity. *Note that a kanban system does not require Pull. There are other ways to limit work to capacity (e.g. explicit WIP limits).*
Ready for Pull (Doing/Done)	Extra columns within each column (where it makes sense).	Usually used when a handover is necessary. Useful to surface queues, indicating bottlenecks.
Definition of Done (DoD)	A short checklist at the bottom of each column (*exit* criteria: things that should be done before the work can be moved to the next step).	To share a common standard for the team. Tweaking the DoD increases quality and consistency of the work.

Blocked Work Items	An extra pink (small) sticky on a work item sticky.	To indicate that the team cannot complete a work item due to dependencies to others (groups, partners, etc.). These should attract attention to be removed a.s.a.p.
Avatars	Magnets with the team's member's initials or symbol.	To see the team's current focus (in real time). Is the team focusing on the right things? May also be used to limit WIP per team member (each team member has max X avatars).
Due Dates	A date on the top left corner of the work item sticky.	To see the promised due-date of a work item. Not all work items should have due dates! Few items with due dates makes planning more flexible, meaning that the due dates will probably be respected. Many items with due dates makes planning brittle, increasing the risk of late delivery.
Classes-of-Service	Different swimlanes on the board. *E.g.* ● *Expedite lane* ● *Due dates at risk lane* ● *Standard lane* ● *Intangible lane*	To understand the risk profile of a work item and treat it accordingly. A high risk profile should attract more attention than a low risk profile (are the avatars correctly places?), meaning that low risk profile items will take more time to reach completion.
Incoming (*optional*)	A buffer named 'Incoming', positioned at the left of the 'ready' column.	To see new customers' requests (new since the last planning meeting). Allows the team to decide if each request is to be worked-on directly, placed somewhere in the ready queue or placed in the 'backlog' for later.
Definition of Ready	A short check-list at the bottom of the 'Ready' column (*enter* criteria).	To see and understand the interface the team has with the upstream partners. It states what the team needs to be able to work efficiently/effectively on a work item (information, resources, etc.).
Explicit WIP Limits	An explicit limit of work in progress. For the whole board (preferred), by column (preferred), by team member or a mix of those.	To put pressure on the system by surfacing things that are in the way for the work to flow smoothly (engine for improvement).
Capacity Allocation	An explicit limit of work in progress per work type, customer, lane, etc.	To divide the available bandwidth (WIP limit) by work type, customer, lane, etc.
Daily Planning Meeting		To ensure that the team is doing the smartest thing right now.

KICK 7: Present a Common Language

	Impact	Can it be skipped?
KICK 7	Outcome: To introduce a language/vocabulary (to use when building a board) that is common to a whole unit/ department/ company. As a result, different teams use the same words for the same things (e.g. everyone knows what the "Prepare" stage means in a workflow).	If you do not have many teams to harmonize or are missing a common vocabulary.

You present a common language to be used by the team when building its board and policies.
This common language may be decided at a company, unit, program or project level (e.g. we develop "features"; the step when the customer validates the work is always called "Get Customer Acceptance", etc.).

Here you can present common visualization and way-of-working policies that the team must use or refer to. E.g. A program may decide that all projects should use a specific visualization scheme as well as a specific escalation policy for issues.

System Development

All development activities within Sandvik IT must comply to Sandvik IT's development process.

The system development process defines the following stages:

1 Prepare for Development
2 Develop Feature
3 Verify with the Customer
4 (Wait for Release)
5 Get Internal Acceptance
6 Get Customer Acceptance
7 Deploy
8 Done

How to use: if the team has a step in its workflow that is about "preparing stuff to be developed", this step should be called "Prepare for Development".

Application Management

All development in the scope of Application Management (AM) should use the terminology defined by the AM processes within Sandvik IT.

- Incidents
- Problems
- Change Requests

KICK 8: Set Visualization Policies

	Impact	Can it be skipped?
KICK 8	Outcome: To agree how on to visualize work in order to understand what is going on and take appropriate smart decisions. Output: a list of policies governing the visualization of the workflow (so that they can be remembered and shared with others).	If the team already has visualization policies that do not need to be revised.

You help the team to discuss and agree on how to visualize the work.
Note that most of these policies are often already decided for you when using a digital board.
Have the team's "secretary" take note of each policy on a sticky and place it on a flip-chart.
The flip-chart will need to be physically attached to the board so that the team can refer to these policies any time.
When using a digital board, try to enter the policies on the board (e.g. use the comments available for each column when using Lean Kit Kanban).

Work Types

How to visualize a Work Type?
Ask the team to select a sticky color for each of the *work type* identified earlier.
Ask the secretary to write down the policy.

This may prove the most difficult part of the workshop :-)
When possible, use stickies with very contrasting colors as some team members may have some sort of color blindness (up to 10% of Caucasian males).

Work Items

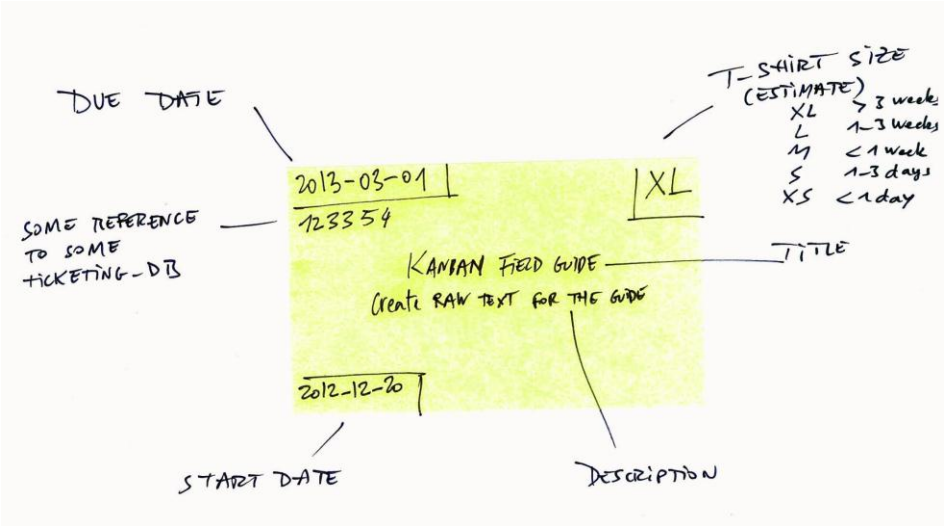
How to visualize a Work Item and what information must always be present?

Propose the following default Work Item visualization:

- *[Top-left]* **Due Date** (if any).
- *[Top-left]* Some **reference** to a ticket database or equivalent (if any).
- *[Bottom-left]* The **Start Date**: date at which work was started by the team on the item (when a team member pulls the item from the "Ready" column, or a manager "pushes" the item onto the board).
- *[Top-right]* An **estimate** on the amount of work represented by the item. This is a man-time estimate (how long time would it take an average team member to bring this item to "done", if the team member were to focus 100% on the item).
  Estimates are usually wrong! But they can be useful for prioritization; for understanding how much work the team is actually managing; and for giving some sort of indication when an item might be done (in combination with workflow metrics). To minimize estimation waste, we recommend a fast estimation method using T-shirt sizes that only requires a couple of minutes from the team.
  The T-shirt scale goes like this:
  - XL > 3 weeks work
  - L 1 - 3 weeks of work
  - M < 1 week of work
  - S 1 - 3 days of work
  - XS < 1 day of work
- [Bottom-right] The **Stop Date**: date at which the item was completed (reached the "Done" column).

Ask the team if any more information is <u>really</u> needed.

Ask the secretary to write down the policy.



Workflow

How to visualize the team's workflow?
If it is not already done, **ask the team to draw the steps identified earlier as columns on the board.**

Note that some of the columns may be removed later on during the workshop and be replaced by exit criteria in a Definition of Done (more information on that later).

What work NOT to visualize?

By default, all work should be visualized on the board (i.e. be represented as a work item). Is that Ok?
Ask the team what work should NOT be visualized.
Ask the secretary to write down the policy.

Why is this work not visualized? Is there another way to keep track of how much effort in going into that kind of work anyway?

Examples of work that a team may decide not to visualize:

- *Administrative overhead. E.g. meetings, especially meetings that are not related to the ongoing-work (Department meetings).*
  Note that the team may get value in visualizing the week's planned meetings to 1) get a feeling of how much time all these meetings will take from actual work, 2) decide what meetings to prioritize and not prioritize, and 3) off-load busy team members by distributing the meetings to other team members.
  Meetings can be visualized as stickies on a dedicated row at the bottom on the board where you have drawn the days of the week.
- *Work that is short-lived. Some work is so short-lived that the cost of recording and managing it on the board out-weight the value from getting it visualized (e.g. "5 minutes" requests or incidents).*
  Note that this work is now invisible and may steal capacity from the prioritized work. Therefore, it may be interesting to visualize this 'volatile' work by - for instance - counting how many "5 minutes" work items the team handles per day.

How often must the visualization be in Sync?

The team should meet regularly (default is once per day) in front of the board in order to take smart decisions. This requires the board to be in sync with reality.
Ask the team how often they need the board to be in sync.
Ask the secretary to write down the policy.

Examples:

- *A development team dealing with new development should have the board in sync at least right before the daily planning meeting.*
- *An Application Management team dealing with change requests, incidents and problems should have the board in sync anytime an incident occurs.*
- *A team of incidents managers dealing with a constantly changing environment should always have the board in sync.*

KICK 9: Create & Populate the Board

	Impact	Can it be skipped?
KICK 9	Outcome: To visualize the work currently handled by the team, according to the existing policies. Output: a physical board full of colorful stickies, or a digital board full of work items.	If the team already has a board that visualize all on-going work items according to the current policies.

You help the team to create their board and populate it with the team's current work, according to the existing policies.
There is a huge advantage in having the team populate their board as soon as possible: seeing and working with real work items make the policy-related discussions much more concrete and far less academical (which tend to happen a lot when you have a room full of engineers).

Have the team's "secretary" take note of each policy on a sticky and place it on a flip-chart.
The flip-chart will need to be physically attached to the board so that the team can refer to these policies any time.
When using a digital board, try to enter the policies on the board (e.g. use the comments available for each column when using Lean Kit Kanban).

Create & Populate the Board

Draw the workflow

Draw the workflow on the board (as agreed during KICK 4 and KICK 7).
The board should start with a "Ready" column and end with a "Done" column.
Use stickies for the headers of the columns as you will want to quickly reposition the headers to fit the amount of work items in each column when populating the board. Also, do not draw the line between the columns yet; wait until all the work items are in the right column.

Create Work Items

Based on the visualization policies agreed on earlier, ask the team to create work items: one work item per work that this currently in progress (being handled by the team) or that the team has committed to (but may not necessarily being worked on right now).
This is the only moment during the workshop when the team members are allowed to use their laptops to look at mails, open tickets, project plans and other databases in order to remember everything that is ongoing. This step may take a while.

Populate the Board

Ask the team to place each work item on the board according to its status.
Some work items will be difficult to place on the board as their status does not map the work flow (yet). Place these items in the "Ready" column for the moment.
Work items may also change position soon as we will add policies to the board that will affect the locations of work items.

Go through each Work Items

Ask the team members to briefly present the stickies on the board.
Now that the whole picture about what is going-on is in place, it is important for the team to grasp it. So, take time to check each work item, but try to avoid getting stuck in detailed discussion: the team will have time for that later on. The focus right now should be on making sure that the right information is on the board (there may be some duplicates, some invalids, some missing, some not at the right place, etc.).

Reflect on what you see

Ask the team to take a step back and look at what they see on the board.

- *Is that the right amount of work for the team?*
- *Are there any obvious bottlenecks or overloaded areas of the board?*
- *Are there areas that are not used at all?*
- *What does that tell us on how the team is currently working and the chances of the team to deliver something of quality to the stakeholders?*

Make Sure the Work Items are at the Right Place

Create a Definition of Done (DoD)

For each column on the board, ask the team to create a Definition of Done (DoD).
Ask the secretary to write down the policy.

The DoD can be seen as exit-criteria: some conditions that must be completed for a work item to be eligible for being pulled into the next column (step) on the board. The DoD is usually a list of items that must be created or actions that must be completed on the work item.
When all the columns have a satisfactory DoD, ask the team to update the position of each work item on the board to match the new definition of the columns.

Remember: keep it very simple! No matter how long you spend creating a DoD it will be wrong: the team needs to see it in action - daily - on a variety of work items before coming with something actually usable.
The goal here is for the team to discuss and agree on a common way of working. This is the key to creating a Standard for the team that can be improved upon: a requirement for a sustainable continuous improvement.

If the workflow has too many steps (there are too many columns) try to reduce them by collapsing a column into a Definition of Done: the column becomes an extra exit criteria of the column 'before'. Similarly, put focus on critical exit criteria by creating a column for it (e.g. a "validation" or "code review" exit criteria).

Working with a big team? Divide the team into smaller groups (3-6 persons) and give responsibility to each group to create the DoD for a couple of columns on the board. Time box the exercise to 10 minutes per column. Have each group present and explain their DoD. Facilitate a discussion with the whole team to agree on the final version of the DoD.

Create Ready for Pull columns

Ask the team to divide the columns that are not a buffer or queue into a "Doing" and a "Done" column.

Ask the team to update the position of the work items in those columns.
Place each work item in the "Doing" column if it is currently being worked or in the "Done" column if the work is done according to the column DoD but it is not yet pulled into the next column.

Tag blocked items

Ask the team to tag all the work items that are currently blocked (physical board: using a little pink sticky on top of the work-item sticky).
A blocked work item is a work item that cannot be work on by the team due to a dependency to another team, group, individual, partner, etc. that is outside the team.
The blocked tag should describe the reason for the block and the date (time?) when the work item got blocked.

A team member waiting for another team member is not considered to be blocked: the team members should be able to resolve that conflict internally. Still, some large teams may have need of visualizing these internal blocks but must use another color tag than pink (e.g. orange).

Ask the team how to handle blocked work items?
Ask the secretary to write down the policy.

Blocked work items are a symptom that the team's context and policies are not adapted to delivering things fast and smoothly. Blocked items are bad! The blocks must be removed a.s.a.p.

Each block should trigger two discussions:

- *How to remove this block a.s.a.p.?*
- *What new policies must we have in place in order to not be blocked the next time we have a similar situation?*

Reflect on what you see

Ask the team to take a step back and look at what they see on the board.

- *Now that we see all work that is blocked or queuing (waiting to be pulled) how much of the work in progress can be acted upon (actionable)?*
- *What does it tell us about the problems your team may have to deliver something of quality to your customer?*

Prioritization Policies for Work in Progress

Due Dates

(If this wasn't done earlier) **Ask the team to write on the work items any deadline or due date that *must* be respected.** The dates must be written at the correct place on the work item, according to the visualization policy.

These due dates must be promises that have been discussed and given to the customers; not just wishes from either the customer or the team members.

Due dates and urgency: the urgency of a work item with a due date will evolve over time. If the due date is sufficiently far into the future with respect to the item's estimated size, the team treats the work item as "normal" work (like "a.s.a.p."). But, depending on how the work is going and depending on how many exceptions or higher urgency work the team is handling, we may reach a point when the team understands that it cannot deliver the item in time if it keeps treating it as a "normal" item. To deliver in time, the team needs to treat the work item as "urgent" (see next step in this feature for more info).

The number of due dates should ideally be kept to a minimum, as a certain level of flexibility in what work to focus on - or not - is necessary in order to deliver to promises. Therefore, the team will not be able to keep any promises if all work items have strict due dates

Prioritize on-going work

There is a great risk that the board is full of work items, far too much for the team. So, the team needs help understanding what to focus on. Too much WIP makes it hard for the team members to see what is important (what has the highest risk or urgency). To help the team to grasp the situation you can introduce swimlanes for different urgency level.

- Create a swimlane called "*Urgent*" (1 sticky high) - for high urgency work items that require the whole team's attention (incidents in production, things that were promised for yesterday).
- Create a swimlane called "*High*" (2 stickies high) - for work items that are at risk (risk to be late).
- Create a swimlane called "*Normal*" (as high as possible) - everything else, including "a.s.a.p." work items.
- Create a swimlane called "*Low*" (1 sticky high) - for work items that can be delayed without problems (e.g. improvements, long-term activities)

Ask the team to place the work items in the right lane, based on the information collected so far.

This is a simplified class-of-service mechanism with classification by cost-of-delay. As these concepts are quite complex, they will probably not stick with the team during the Kick-start, therefore try to avoid to overload the team members with it just now.
The team may need another prioritization scheme once it is in control over its WIP (when the team feels that the number of work items in progress is aligned with the team's capability).

Notes that, depending on the team, there may be smarter uses for the swim lanes. For example, it may make more sense for the team to visualize the work grouped by customers, systems, company or market risk segments.

Prioritization Policy

Ask the team when a work item should change urgency (e.g. from “Normal” to “High”, or from “High” to “Urgent”).
Ask the secretary to write down the policy.

Due dates at risk should probably be placed in the “High” lane. The question is: when does it become “at risk”? Items that will definitely miss their due dates if continuing as normal should probably be placed in the “Urgent” lane.

Ask the team how to prioritize work within a lane.
If the “normal” lane has many ongoing work items, what item should the team focus on finishing first?

What goes first:

- Oldest items?
- Smallest items?
- Items from customer/system X first?

A good start is for the team to focus on the oldest items first, as these items were - a while ago - prioritized highest by the customer.

Definition of Ready

Ask the team to define the characteristics of a work item that is ready to be worked on by the team.
Ask the secretary to write down the policy.

A good Definition of Ready helps avoiding that the work items become blocked later in the workflow, as it guarantees that the work items have the necessary properties to be fit-for-work.

Typical elements of a definition of ready could be:

- *There is a contact person.*
- *The contact person is available when we are about to start working on the item.*
- *There is some sort of specification of the demand, possibly in a written form.*
- *There is enough information to understand what it means for the work to be “done” (validation criteria).*
- *There is some sort of billing information (if applicable/necessary).*

KICK 10: Set Way-of-Working Policies

	Impact	Can it be skipped?
KICK 10	Outcome: To agree on how to handle work. So that all team members handle work in a consistent way. Output: a list of policies governing the way work is handled.	If the team already has way-of-working policies that do not need to be revised.

You help the team to discuss and agree on policies governing its work.
Have the team's "secretary" take note of each policy on a sticky and place it on a flip-chart.
The flip-chart will need to be physically attached to the board so that the team can refer to these policies any time.
When using a digital board, try to enter the policies on the board (e.g. use the comments available for each column when using Lean Kit Kanban).

Manage the Demand

Ask the team who is managing the "Ready" buffer.
Ask the secretary to write down the policy.

Based on the demand on the team, this person prepares the work items to be worked on, places them in the buffer and may prioritize them. In effect, this person decides what the team does and in what order.

This does not need to be one person. Actually, it is better if the team itself is in charge the demand.

Ask the team when and how the "Ready" buffer is replenished and re-prioritized.
Ask the secretary to write down the policy.

This question may be quite tricky to answer. If the team lives in a complex environment (multiple upstream partners, possibly with different profiles, or multiple systems, etc.) the team may need to set in place a set of policies to be able to treat the demand consistently. To answer this question thoroughly is out the scope for the Kick-start workshop. This is addressed after the Kick-start in "After the Kick-start: Reach Continuous Improvement Baseline" chapter.

If the team manages its own demand, here is a simple policy that can help: use an "Incoming" buffer. Create an "Incoming" buffer to the left of the "Ready" buffer. Team members should use the buffer to place potential new work items that are new since the last planning meeting. At the start of the next planning meeting, the team decides how to handle each new request. The request can be:

- *pulled in directly (because it is an expedite or the due date is at risk),*
- *placed on the ready queue (it is at least as valuable as the other X items in the queue),*
- *placed in the 'backlog' for later, or*
- *simply discarded?*

Ask the team when, how and by who is the "done" column managed.
Ask the secretary to write down the policy.

This is especially important if the completed work must be forwarded to downstream partners.

Side Orders

Ask the team how to handle demands coming "from the side" (i.e. not via an official channel: chat, telephone calls, "corridor" orders, etc.).
Ask the secretary to write down the policy.

This can be a complex problem to solve as you do not want to alienate your customer while creating time to work on the right thing. A right approach for dealing with these situations is for the team members to help the customer help herself so that she does not need to re-contact the team the next time the same situation occurs. Another way

would be to place a sticky for the request in the "Incoming" buffer and tell the requester "Thanks for your request! We will discuss and schedule it during our next planning meeting tomorrow morning. I will come back to you after the meeting to tell you how we will handle it".

When to Pull?

Ask the team when and how the work is pulled from the "Ready" buffer.
Ask the secretary to write down the policy.

- *A solid policy would state that: a team member should pull new work in only when he/she cannot contribute to any other ongoing work.*
- *If you are using WIP limits, the policy should state that worked is pulled only if the WIP limit is not violated. Violating the WIP limit should trigger a discussion ""Is it Ok? Is the WIP limit actually too low? Etc.".*
- *An excellent policy would also dictate that instead of violating the WIP limit and pulling new work, a team member should work on intangible work (e.g. improving test environment, compilation scripts, documentation, refactoring, etc.).*

Part-timers

If some team members work part-time, **ask the team how to organize the work of the team-time members.**
Ask the secretary to write down the policy.

- *Should the part-timers attend all planning meetings? Only the days that they are working for the team?*
- *Is it Ok for the part-timers to spend time in the team anytime or should it only be certain time of the day (mornings!), certain days of the week (Thursdays and Fridays)?*

Other Policies (already set earlier)

Other Way-of-Working policies that have already been set earlier are:

- Definition of Done
- Definition of Ready
- Blocked Items
- Urgent Work
- Due Dates at Risk

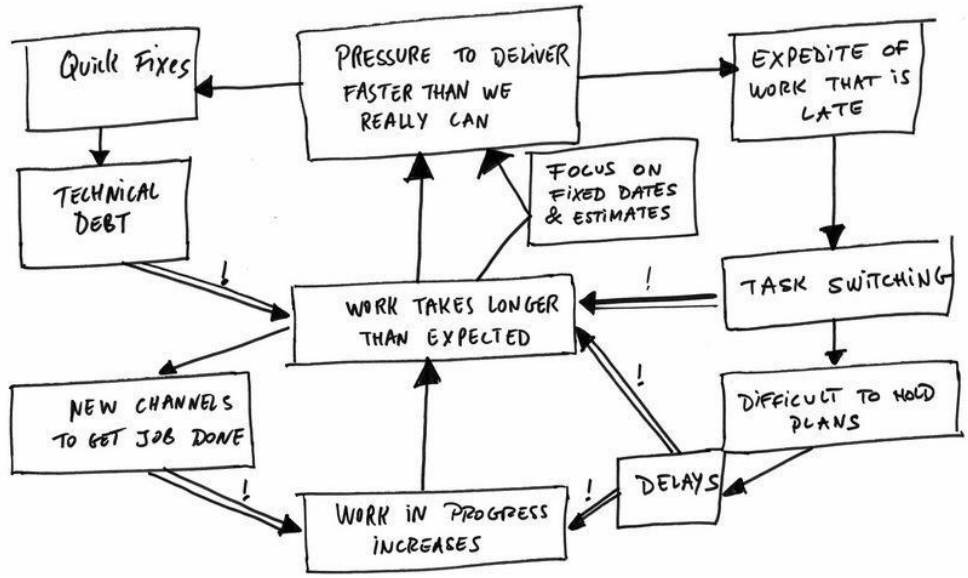
KICK 11: Limit WIP

	Impact	Can it be skipped?
KICK 10	Outcome: To agree on how much work is currently optimal for the team. Output: a policy dictating how much work is allowed on the board (per column, swimlane or person).	If the team already has a satisfactory WIP limit policy, OR the team isn't mature enough yet to sustain the policy.

You help the team to limit the work in progress.

Uncontrolled Flow

Discuss what happens when the flow of work is uncontrolled.
Use the following causes and effects diagram to explain the impact of too much WIP. This diagram was created based on the experience of 20+ development teams at Sandvik IT.



To Value Finishing over Starting

Based on the negative impact of uncontrolled flow, the minimal policy that each team must commit to should be:
Commit to finish work over starting new

Strategies to Reduce WIP

Discuss with the team how to reduce WIP.
Here are some ideas:

- Breakdown large work items (L or XL) into smaller ones
  This can easily be done by having a DoD for some early "prepare work" stage that states that work items must be broken down to, say, maximum M size.
  Large work items abstract a lot of work as well as a lot of risks. An item that stays several weeks on the same spot does not give much information. And when it is blocked, is everything blocked or just a part of it?

- Focus on reducing blocks
  It usually works like this: you pick a work item, start working on it and then realize that you need some other guy, from some other team, to give you some information. You mark the work item as blocked and then pick a new work item. Which gets blocked. Which forces you to pick a new... Therefore, focusing on removing blocks and reducing their occurrences will help you keep down the amount of ongoing work.

- Use Pull
  A pull mechanism ensures that new work items are only started when a team member has capacity for it. Therefore - eventually - the team will reach a stable level of WIP that matches the team's current capability. That is to say that the team members will not be stressed, shortcuts will not be taken, there will be a reasonable amount of task switching, etc. One can say that the team reaches a "natural" level of WIP and usually can maintain it.
  Note that this only works if the team uses Pull consistently. The WIP increases anytime work is "pushed" into the team. This can happen when some incident occurs, when some urgent work is discovered, when customers set aggressive due dates, or when a manager just wants things to happen. The only mechanism for limiting WIP in push situations is explicit WIP constraints.

- Limit WIP explicitly (WIP constraints)
  You explicitly set limits on the amount of work the team is allowed to work on at any moment. The immediate benefit is to limit work in progress to the team's current capability. The long-term benefit is to create a *constraint* that forces the team to devise new policies to be able to handle less on-going work (i.e. less blocked work, minimize the size of queues or remove them all together, better management of the demand, etc.).
  Explicit WIP limit is one of the engines for improvement built in Kanban, especially when you lower the WIP limit when the kanban system has reached a stable state.

***IMPORTANT**: A team with low maturity (which is probably the case during the Kick-start) will probably not be able to sustain an explicit WIP limit, even if the whole team "gets" it during the Kick-start. An explicit WIP limit policy requires most the other concepts discussed during the Kick-start to be understood and followed. Depending on the team current context, experience and maturity, this will take more or less time. Therefore, our advice is to focus the team's energy on getting the basics right, use a Pull policy and avoid WIP limits at first. This means that explicit WIP limits must be introduced later on (see BOOST 4).*
If the team has too much on its board right now, the first action is to reduce the WIP. One, usually painless, way to reduce WIP is to use a Pull policy to reach a sustainable WIP (this may take several weeks or even months). Another, more disruptive way, is use a Pull policy and to eject most of the ongoing work to reach a more "normal" level of WIP. Decide on a "good" level of WIP (look at the "Set explicit limits" section in this feature) and ask the team to eject work items until the agreed level is reached (ejected work items go back to the "Ready" column).

Set Explicit WIP limits

Discuss where to set the explicit WIP limits.
Explicit WIP limit can be set in a number of ways. Some are more aligned with the spirit of the Kanban method as they effectively drive the team's improvements; others help limit work to capacity without constraining the team to evolve.

- Limit WIP per column on the board
  Each stage in the workflow may have its own WIP limit.
  E.g. "Prepare" has a limit of 3, "Develop" a limit of 5, etc.
  A well-functioning set of WIP limits per column can ensure that there are an optimal number of work items for the mix of specialists in the team. The most Kanban-aligned approach is to have the WIP limits cover all work items in a column (work items being worked, work items blocked and work items in queues).Tweaking the WIP limits can improve flow in the kanban system.

- Limit per swimlane or Work Type (i.e. capacity allocation)
  Each swimlane (if relevant) may have may its own WIP limit.
  Some team may have a swimlane per class-of-service, customer, system, or other.
  E.g. "Urgent" has a WIP limit of 1, "Customer X" has a limit of 5 while "Customer Y" has a limit of 10, "Change Requests" work type has a limit of 3, etc.

This model allows to allocate the team's capacity to certain work types, stakeholder or market risk segments. Tweaking the WIP limits can improve flow in the kanban system, but may require management to be involved in the decision (as changes have a direct impact on the business).

- Limit per person
  Each team member may have its own WIP limit.
  E.g. "Kalle" has a limit of 4, "Olle" a limit of 2 (he works part-time).
  This model is good for controlling multi-tasking but does a poor job controlling the total amount of work in the system (work stuck in queues for example). In practice this model has more in common with a personal Kanban board rather than a real kanban system. Tweaking the WIP limits can improve team members' situations but does little to improve flow in the kanban system.

- Global limit
  The kanban system has a total WIP limit, regardless of where the work items are.
  E.g. the WIP limit for the whole team is 20.
  This model has the advantage to include all queues and buffers in the limit. This makes it the model that most appropriate to improve the system, though it may be the most challenging for the team.

- A Mix
  These different strategies can be combined for best effect.
  For example: A team may have a global limit of 20, as well as a limit of 10 for the "Application Management" swim lane, and a limit per person of 3.

Discuss what WIP limits to set

- Establish a global target WIP limit for the whole kanban system.
  Here are some strategies on what WIP limit to start with:
  - Use the current WIP level (this is probably too high, though the WIP limit will be trimmed and will - eventually - become "right").
  - Use any number that feel reasonable to the team (this is arbitrary, but this number will be trimmed).
  - Use the number of team members x 2 (aggressive) or 3-4 (comfortable)
  - Use the "natural" WIP level: start the without WIP limits but use a Pull policy with the focus on finishing work over starting new. After a (maybe long) while, the system should naturally balance itself to some WIP level. Use this as starting WIP limit.

- Allocate the global WIP limit:
  - To columns.
  - To swimlanes or work types.
    The team manager should help during the allocation in order for it to be aligned with the team/division/organization objectives.

Discuss what to do when a WIP limit is about to be violated.
How should the team behave when a WIP limit is about to be violated?

WIP limits exist to make clear our assumptions on the current ideal level of WIP for the team. The current ideal level may very well change under time depending on new context and information. Violations of the WIP limits give us valuable information that we must use to validate our assumptions. Therefore, violations must trigger a discussion.
The question to resolve is then: What do we do? Should we temporarily allow the violation because it is an exception? Is it obvious that WIP limits are too low and should be raised? Should we not allow the violation? etc.

Discuss how and when to trim the explicit WIP limits.
In Kanban, the WIP limits act as constraints that force the team to improve its processes and way-of-working. When the team is comfortable working with the current WIP limit, it may be time for decreasing the WIP limit by some units to force the team to discover new creative ways to cope with the reduced WIP. This may trigger new policies to deal with blocks, queues and delays (e.g. how to deal with a customer that does not have time to validate work, how to deal with a slow building process, how to deal with manual regression tests, etc.).

KICK 12: Set Planning Meetings Policies

	Impact	Can it be skipped?
KICK 12	Outcome: To agree on how to use the board to take smart decisions. Output: a policy dictating when to meet, how (especially for distributed teams), who holds the meeting, the purpose of the meeting and its details.	If the current meeting policy exists and is satisfactory.

You explain what the Planning Meeting (purpose and setup) is.
You guide the team in deciding policies governing when, how and where the meeting holds place.

The Planning Meeting 101

Explain the need for a planning meeting

- For the team members to see and understand what is going-on.
- Based on this knowledge, to have the team do the right thing (right focus).

The purpose is to plan the day, not to report status (the board is already doing that).
The board is not what is important during the planning meeting: the discussions in front of the board are what is important.

Set meeting policies

Ask the team when they need the board to be in synch with the real world.
Continuously, once a day just before the planning meeting, several time per day?

Ask the team who is the facilitator for the meeting.
The same person every day, a new team member every day, week, month?
A good policy is to have the team members taking turns in facilitating the meeting. This will make it easier for them to understand that they own the board and it allows them to look at the board from a slightly different perspective (and finding new potential improvements when doing so).

Ask the team if they can have the meeting every day.

- **If no, how often can they have the meeting?**

The more time between meetings, the more needs to be synchronized (takes time) and the more exceptions (risks, opportunities, re-prioritizations, etc.) are left unaddressed (with potentially negative effects that grow exponentially with time).

Ask the team at what time the meeting can be held.
Most of the teams are most comfortable with planning meetings early in the morning, in order to plan the day. When working distributed, find a time that fits all team members. If it is not possible to find a reasonable time for everyone (e.g. a team with members in the US, EU and Asia), try to have two meetings a day (e.g. for team members in the EU: first a meeting EU-Asia in the morning, then a meeting EU-US in the afternoon. The team in the US will then have a meeting US-Asia, etc.).

Ask the team Where to meet
At the board if using a physical board. Where will the board be?
For distributed teams: in a conference room for the team members that are collocated, or each team member in front of his/her PC?

Policies during the meeting

During the meeting, the team should follow the following policies:

- Run the script/agenda for the meeting (look at KICK 13)
- The meeting takes maximum 15 min!
- The meeting is for planning only: issues resolving should happen during the “after meeting”.
- If you find that the meeting gets stuck in some (potentially boring) conversation: Rise your hand! (The idea being that the person speaking will stop when he/she sees a forest of hands - hopefully).

The After Meeting

When the planning meeting is done (according to the agenda), the team should have time to discuss in details any issues.

Recommended Planning Meeting Policies

- The team **updates the visualization board at least once per day**, and at least before the planning meeting, to reflect the current situation.
- **The team meets every day** at the same time.
- **All team members participate** (either physically or via video- or teleconference) to the meeting, even if some the team members only work part-time, even if some the team members are not on-site.
- All team members have **access to the visualization board** during the team’s planning meeting (either physically, via photo, video or an online digital version of the board).
- The team’s planning meeting ends when 15 minutes are up regardless of how much is left to discuss (planning meeting **is time-boxed to 15 minutes**).
- The team’s planning meeting is **facilitated by one team member** (not necessarily by the flow manager and not necessarily by the same person every day).
- **An after-meeting** is held after the team’s planning meeting if there are any remaining issues worth discussing, where only the team members concerned by the issues are asked to stay.

KICK 13: Run the First Planning Meeting

	Impact	Can it be skipped?
KICK 13	Outcome: For all team members to understand how the meeting works. Output: the first planning meeting is done.	If all team members are well used to (daily) planning meetings.

You facilitate a planning meeting for the team by taking it step-by-step and very thoroughly.

Planning meeting for dummies

Act as facilitator for the team's first planning meeting.
Run the script with the team.
This script is intentionally simple as it is intended for green teams. For instance there is no WIP limit check here. The team's policies discussed earlier override this script. A team will (must!) update this script whenever it is needed.

Take your time; this first meeting will take more than 15 minutes... At each step, ask the team members if they have understood what you did and if they have questions. Repeat each step twice.

The Planning Meeting Script

1. **Check status**

Is the board in synch? - Update board to reflect the current status

2. **Read the board**

What's **new** since last meeting? - On the board, in the "incoming" column?
What do we see? - Are there bottlenecks? Overloaded areas?

3. **Plan**

What are the **most important** work items right now? - Urgent? Date-at-Risk? Oldest?

- Who is working on them?
- Is the work item blocked?
  - What can we do to un-block it?
  - Who is in charge of removing the block?
- Do you require assistance for completing the work item?

Until we have no work capacity left (until next meeting):

- What are the **next most important** work items right now? Date-at-Risk? Oldest?
  - Who is working on them?
  - Is the work item blocked?
    - What can we do to un-block it?
    - Who is in charge of removing the block?
  - Do you require assistance for completing the work item?

4. **Check**

Do we have a plan we believe in?
Are we working smart?

KICK 14: Close the Scene

	Impact	Can it be skipped?
KICK 14	Outcome: the team is now in charge! Output: feedback about the workshop for you to improve.	Always run, except if you *really* have no time left (here is an improvement for you: keep time for closing the scene).

You close the workshop by presenting the outcome of the workshop and what will happen next. Also, you should gather feedback in order to improve the next workshop.

What just happened?

Give a quick summary of what the team achieved during the workshop by going through the outcome and output of each item on the agenda.
The team has now been Kick-started! The team has a board visualizing the current status of all ongoing work items, along with the current policies governing how the work works.

What's next?

Based on the plan you have for the team, explain what will happen now (when you will meet the team again, what will be the purpose, etc.). Look at the features presented in "After the Kick-start" for input.

Deal with the loose ends...

Remind the manager that he/she is in charge of resolving some of the issues discovered during the day (retrospective, parking-lot).

Until next time

Wish the team a good luck and bon voyage!

Feedback time

As a very last activity, ask the team to give feedback on the workshop.
Ask the team members to write some comments (or not) on a sticky. They may choose between a :

- Green sticky (if they think the workshop was great!),
- yellow sticky (if they have mixed feelings or don't care)
- Red sticky (if they think this workshop sucked!)

The team members can leave the comment on the door on the way out.

After the Kick-start: Reach Continuous Improvement Baseline

The team is kick-started! Now is the time to support the team in continuing to build and maintain its kanban system. The new goal is to reach a "continuous improvement baseline": a point where the team is truly aware of its current condition, and can maintain that awareness. This awareness is what is needed for the team to really improve and evolve into what it must.

Definition of Done:

- *The team reached Baseline*
    - *The team's planning meetings have the right outcome.*
    - *The team understands its current capability.*
    - *The team regularly updates, adds and remove policies to fit its current understanding of its role and context.*
    - *The team has policies to manage its demand.*
    - *The team has a strategy to limit its WIP.*

Features

	Feature	Impact	When to Run?
BOOST 1	**Adjust Planning Meetings**	Outcome: The team's daily planning meetings have the right outcome ("do the right thing"). Output: Some input to the team about how to hold better meetings.	Run this directly after the Kick-start. *You may skip this step if the flow manager is experienced.*
BOOST 2	**Adapt Policies to Current Understanding**	Outcome: The team's visualization scheme and policies are adapted to the team's understanding of its current condition. Output: An updated visualization board (columns, items, DoD, policies) and policies.	Within a month of the Kick-start. It is recommended to have attended at least 2 planning meetings with the team prior to running this feature. *This feature can be seen as an extension of the Kick-start workshop. It should not be skipped.* *This feature can be run as many times as needed.*
BOOST 3	**Understand Current Capability**	Outcome: The team understands its capability and can communicate it (lead-times and throughput). Output: CFD and Control Charts regularly updated.	Run this a.s.a.p. after the Kick-start. Skip it if the team already produces CFD and CC.

BOOST 4	**Limit WIP**	Outcome: The team limits the work in progress to its capability (i.e. the team is not under stress). Output: An explicit WIP policy.	Run this when the team can sustain all the other policies and/or the team has reduced its WIP to a sustainable level. Skip this if the team already has explicit WIP limits that are actually in use. *This feature is required for evolving the team's visualization board and policies into a kanban system.*
BOOST 6	**Manage the Demand**	Outcome: The demand on the team is managed following explicit policies in a sustainable and repeatable way. Output: A "Ready" queue replenishment policy.	Skip this if the team already has a "Ready" replenishment policy that actually works. *This feature is required for evolving the team's visualization board and policies into a kanban system.*
BOOST 7	**Assess Depth of Kanban**	Outcome: The team has 1) a better understanding of Kanban, 2) an idea about their current condition, and 3) some input about what to improve to improve their kanban system. Output: An evaluation of the team on the depth of Kanban chart.	You should wait "a while" after a Kick-start, until the team has matured, before running this feature. *This feature is required for improving the team's kanban system.*
BOOST 8	**Set Next Target Condition**	Outcome: The team knows what to problem/issue to resolve in order to improve their capability, in line with their "true north". Output: A target condition.	Run this when the team has become comfortable using its kanban system (following all current policies, new policies are made explicit, planning meetings work). *This feature is the beginning of the next phase: improve relentlessly in a sustainable way.*
BOOST 9	**Give Feedback back to the Guide**	Outcome: The field guide gets better and others can benefit from your insights. Output: Some concrete suggestions are sent back to the ones maintaining the field guide.	Run this whenever you have feedback to give on this guide.

BOOST 1: Adjust Planning Meetings

	Impact	Can it be skipped?
BOOST 1	Outcome: The team's daily planning meetings have the right outcome ("do the right thing"). Output: Some input to the team about how to hold better meetings.	Run this directly after the Kick-start. *You may skip this step if the flow manager is experienced.*

You attend some of the team's planning meetings to help the team conducts the meetings as intended.
Some teams may need many visits from you, others get it directly.

Prerequisites

The team has completed the Kick-start workshop.

Observe and Point-out Problems

Attend the team's planning meeting.
Observe if the team is following its meeting policy:

- Is the team meeting in time?
- Are all the required attendants present?
- Is the board updated before the meeting?
- Is the script for the meeting followed?
- Is the meeting done within 15 minutes?
- Has the team a plan when the meeting is done?
- Are all visible problems addressed or at least discussed?

If not, take the team's attention and point out the problem. Should the policy be updated to reflect the new state, or should the team fix the problem? Allow no more than a couple of minutes for debating the issue. If the issue requires more time to be resolved, propose to postpone the discussion until after the planning meeting.

If the team really has a terrible meeting, propose that you facilitate the meeting the next day. Use that opportunity to repeat KICK 13.

BOOST 2: Adapt Policies to Current Understanding

	Impact	Can it be skipped?
BOOST 2	Outcome: The team's visualization scheme and policies are adapted to the team's understanding of its current condition. Output: An updated visualization board (columns, items, DoD, policies) and policies.	Within a month of the Kick-start. It is recommended to have attended at least 2 planning meetings with the team prior to running this feature. *This feature can be seen as an extension of the Kick-start workshop. It should not be skipped.* *This feature can be run as many times as needed.*

You help the team adapt its board and policies to the new understanding it now has on what it does, how and for whom. Once this is done, you help the team identify obstacles in the way to reach *baseline*.

Prerequisites

The team has completed the Kick-start workshop and has run at least a dozen planning meetings. Under this time, you have made sure that the team runs its planning meetings as intended.

(First Time after the Kick-start) Get Buy-In from the Team

The goal of this step is to **get buy-in** (or buy-out) from the team on their new way of working. The team members get a common understanding on what works better (or worse) now and what needs more attention.

Make the team members reflect on how their situation looks now compared to how it was prior to the Kick-start.

There are many ways to do that, some quick and some more thorough (check "Agile Retrospective" by Esther Derby and Diana Larsen).

- The easiest way is to ask the team members to write down on stickies what works better now and what works worse. Then each team member should present his/hers stickies and place them on a white board.
  This method is actually a bit of a scam as the way the question is asked it creates very few "worse" stickies (and the ones ever coming up are can be formulated in a positive way). But, it works in the sense that it sells the new way of working.
- A slightly better way is to ask the team members to write down and group the stickies according to:
  - Things that you are doing and that you want to continue doing
  - Things that you are doing but want to stop doing
  - Things that you are doing but want to do more of
  - Things that you are not doing but want to start doing

  This usually generates more interesting discussions. The team usually identifies issues that can be resolved by new policies. Some of the issues should be further discussed (create new work items for them), while others should be given to the manager for resolution.

Adapt Policies

Ask the team members whether the visualization policies need to be updated to reflect their new understanding about how they work.
The board (columns, lanes, work types, etc.) may need to be modified based on the new understanding that the team has on its current situation.

Ask the team members whether the working policies need to be updated to reflect their new understanding about how they work.

- Go through all the policies (that are hopefully visualized somewhere).
- Remove the policies that are not in use.
- Add any missing policies.

Identify Obstacles in the Way to Reach Continuous Improvement Baseline

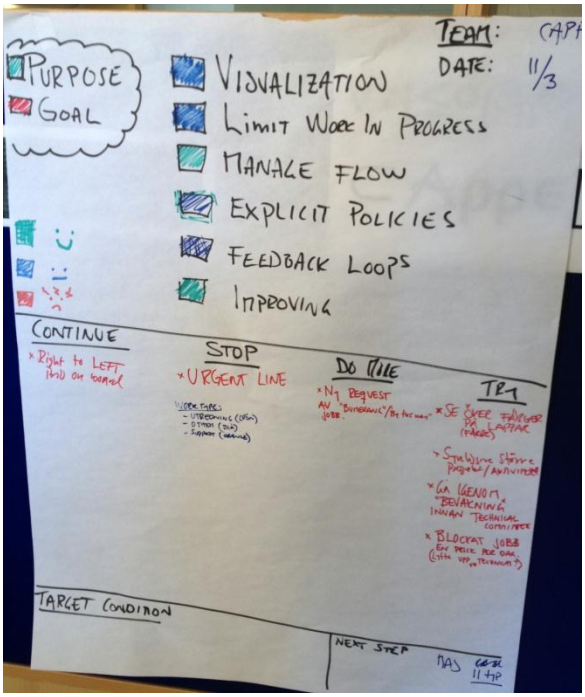
Work with the team to identify what is in the way for the team to reach baseline.

A team is considered to be in control (at baseline) when it is capable of:

- Visualizing all on-going work (according to its policies),
- Having explicit policies that are accepted and followed by all team members, and possibly others outside of the team (note that these policies do not have to be perfect nor "right" - yet),
- Demonstrating the ability to maintain these policies as well as modify them when necessary (add, remove, modify)
- Measuring flow and other KPIs of interest (lead-times, throughput, block-time, etc.).

Is the team at baseline? Celebrate! Else, use the Boosts presented in this guide to reach baseline.

Here is an example on how to use the "Depth of Kanban" (BOOST 7) with the team to identify obstacles and select the most pressing issue to resolve to reach baseline. The coach goes through the different properties of Kanban including "Purpose" and "Goal" with the team to identify the properties that need more work.



Select Most Pressing Obstacle and Remove it!

Ask the team to select the most pressing obstacle.
If the team has identified several obstacles, ask the team members to vote for the most important one.

Ask the team to come up with experiments designed to remove the obstacle.
Ask the team to try the first experiment a.s.a.p. (the next day?).
After some time, check the results of the experiment.

- **If it worked, ask the team to create a new policy to settle the experiment as a permanent fix.**
- **If it didn't work, ask the team to try another experiment.**

BOOST 3: Understand Current Capability

	Impact	Can it be skipped?
BOOST 3	Outcome: The team understands its capability and can communicate it (lead-times and throughput). Output: CFD and Control Charts regularly updated.	Run this a.s.a.p. after the Kick-start. Skip it if the team already produces CFD and CC.

You help the team, and especially the flow manager, to measure the flow of work. This information can then be used by the team to understand its capability.

Prerequisites

The team has completed the Kick-start workshop and you have run SETUP 4 with the flow manager.

Create a Cumulative Flow Diagram

You help the flow manager to produce a cumulative flow diagram.
The diagram should be updated at least once per week.
Contact Operational Excellence Support for an Excel template.
Note that some digital board services create CFD automatically.
You can also check these instructions on how to create your own CFD template (thanks to Håkan Forss).

Create a Control Chart

You help the flow manager to produce a control chart.
The diagram should be updated at least once per week.
Contact Operational Excellence Support for an Excel template. Some digital board services create control charts automatically.
You can also check these instructions on how to create your own Control Chart (thanks to Håkan Forss).

Other Charts

There are many other charts to use for understanding flow (see example below).
One chart that is gaining popularity in the Kanban community is the Spectral Analysis.
Check Håkan Forss instructions on how to build a Spectral Analysis.

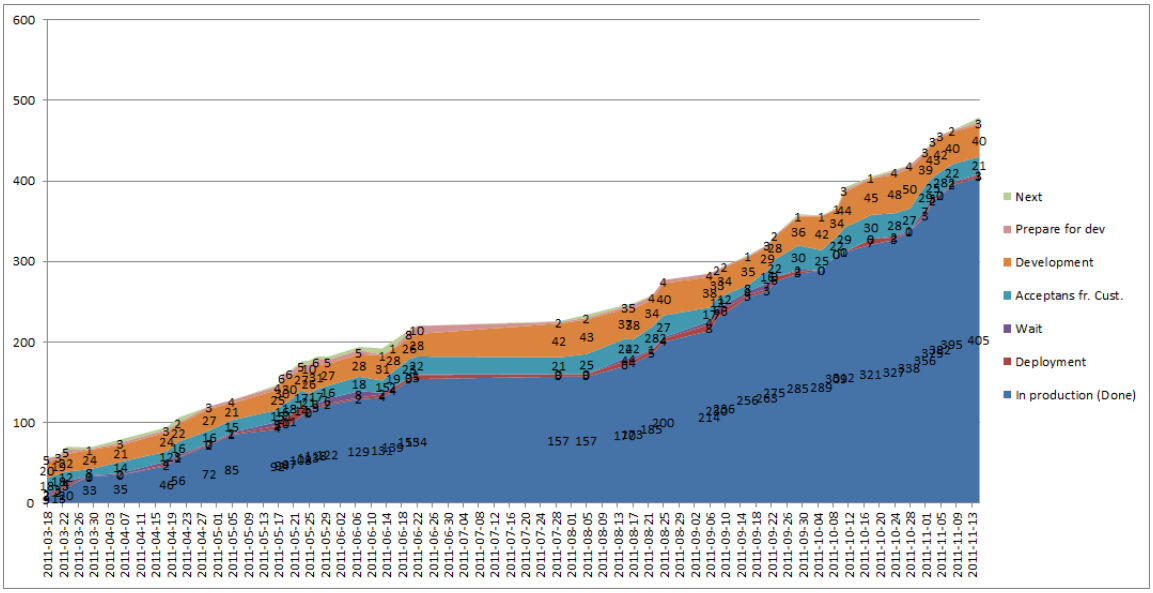
Use the Diagrams to Create Understanding and Start Investigate

The goal is to see trends in the team's capability. The diagrams should mainly be used to understand the team's current condition and to understand the impact of new policies on the team's capability.

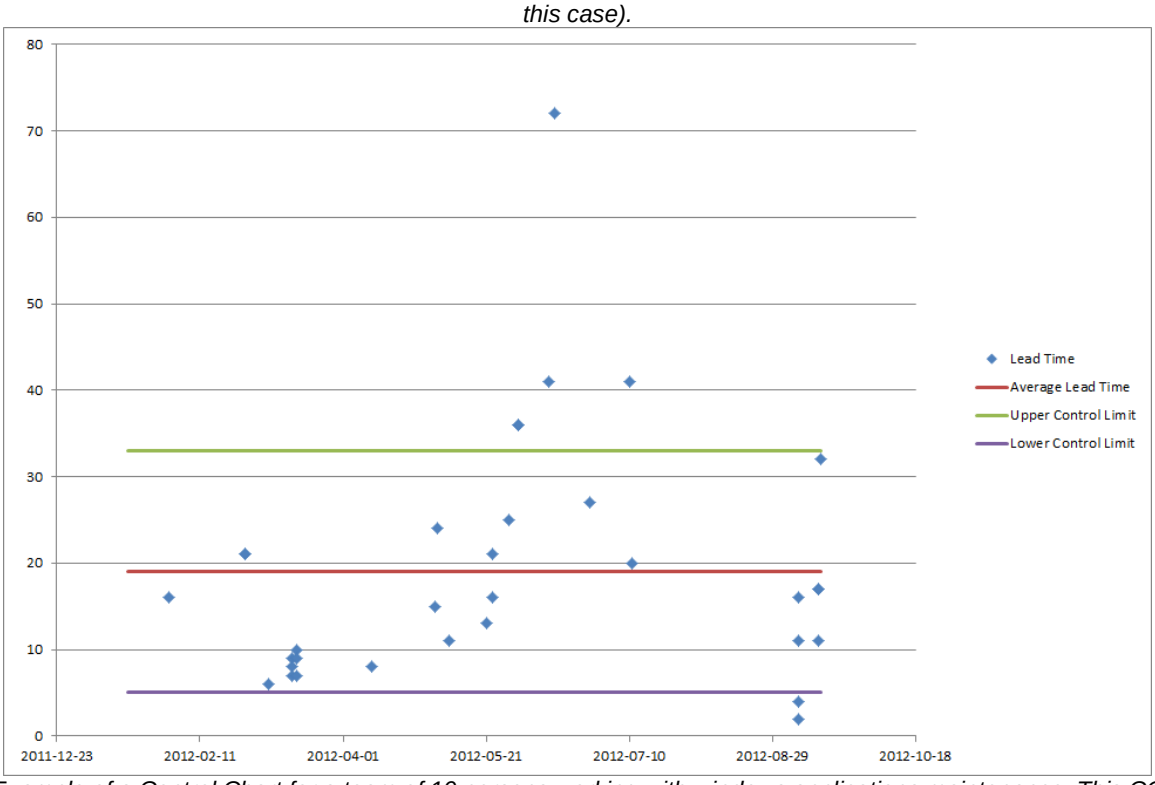
Using CFD: After introducing the latest policy, is the team delivering faster or slower? Why does the "Test" stage has more and more work in progress? For example, using cumulative flow diagram, a team could see that it improved its throughput by 13% in one year.

The control chart can also be used to give promises that the team can keep to its customers. For example, the team could see that it usually delivers "Change Requests" work types of size "M" in 2 weeks +- 4 days (under 2 weeks and 4 days 85% of the time).

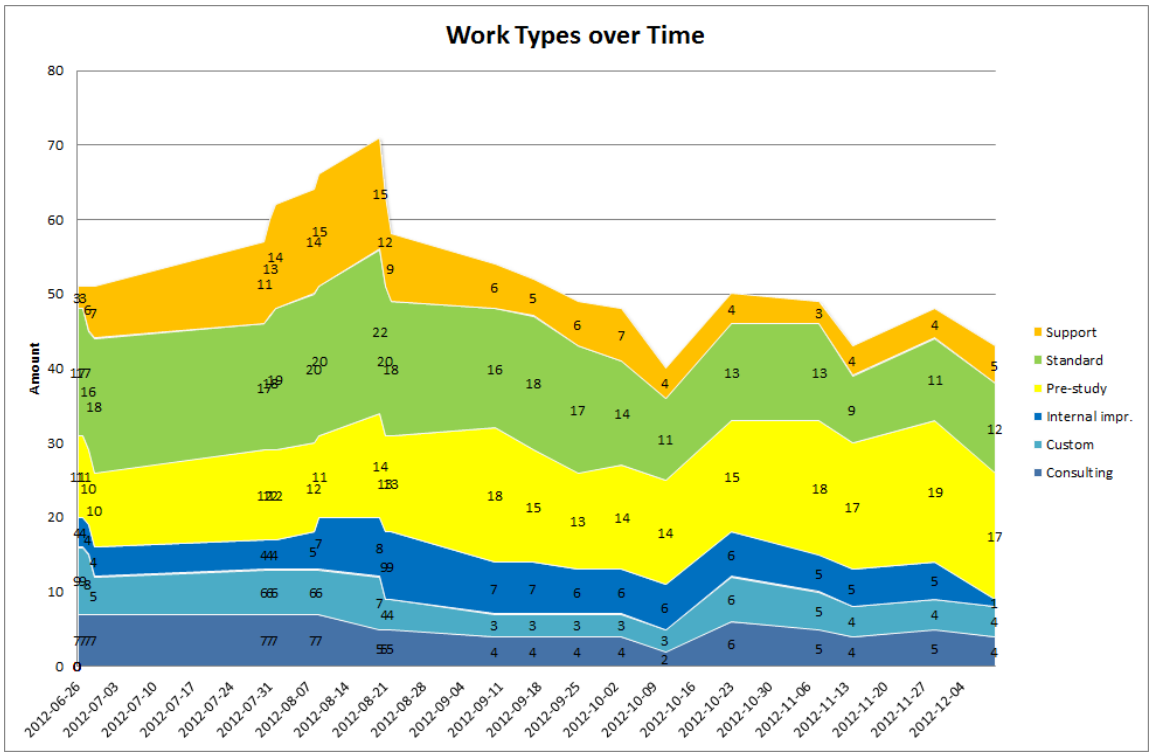
Examples of Diagrams



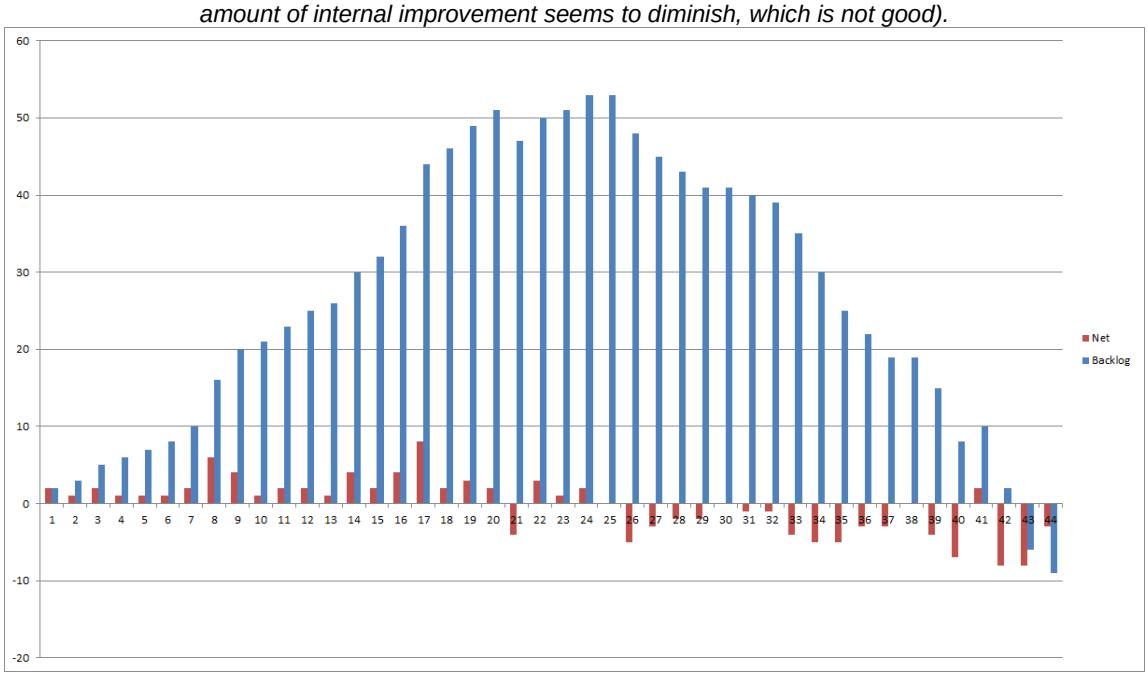
Example of a CFD for a team of 6 persons working with a document delivery product. The flat area in the middle is the summer holidays (they tend to be rather long in Sweden). Note that the slope is steeper by about 10+% by the end of the year meaning that the team has improve its capability, or the context has changed (it was the former in this case).



Example of a Control Chart for a team of 10 persons working with windows applications maintenance. This CC shows the time it took to complete "Small" work items from January to August. Using this graph, the team could tell a customer: "We can deliver an S work item within 33 days with 85% chance, though it can be as low as 19 days for 50% of the items". The area between the upper and lower control limits is rather large, which should trigger the team to analyze the root causes behind the delay for the work items "above" the upper control limit. Fixing the root causes (creating new policies or modifying existing ones) would allow for less variability.



Example of other graphs useful for flow analysis: the "work types over time" graph. This graph shows the evolution over time of the amount of work items on a team's board working with a content management platform. From this graph we can see that the team as a rather constant mix of work items, that the team works to reduce the WIP on its boards though without using explicit WIP limits (the total amount should be rather constant), and that the team has worked to reduce the amount of support issues. The trend seems to evolve in the right direction (though the amount of internal improvement seems to diminish, which is not good).



Example of other graphs useful for flow analysis: the "queue size" graph. This graph shows the evolution of the size of the queue (or backlog) for "standard" work items over the year (the numbers are week numbers). This is the same team as shown on the previous graph. The "net" column shows the difference between incoming and completed work items during the week. The team started to use Kanban by week 24. You clearly see that the team focuses more on "standard" work type from the introduction of Kanban (the net is negative). The negative queue is a glitch of the Excel formula: the queue is simply empty.

BOOST 4: Limit WIP

	Impact	Can it be skipped?
BOOST 4	Outcome: The team limits the work in progress to its capability (i.e. the team is not under stress). Output: An explicit WIP policy.	Run this when the team can sustain all the other policies and/or the team has reduced its WIP to a sustainable level. Skip this if the team already has explicit WIP limits that are actually in use. *This feature is required for evolving the team's visualization board and policies into a real kanban system.*

You help the team to limit work in progress to its capacity.
Depending on the team context and maturity you may have skipped this step during the Kick-start, or the team has dropped the limit WIP policies since the workshop and needs to re-start.

Prerequisites

The team has completed the Kick-start workshop and the kanban system has stabilized (you have run BOOST 2: Follow-up with the team at least once, the team holds its meetings and policies).

Limit WIP

Run KICK 11: Limit WIP

BOOST 5: Increase Flow

	Impact	Can it be skipped?
BOOST 5	Outcome: The team increases the flow of work. Output: A set of explicit policies.	Run this when flow is too slow.

You help the team to increase the flow of work.

Prerequisites

The team has completed the Kick-start workshop.

Strategies to Increase Flow

Here are *some* strategies that can help you increase flow. As context is everything, it is not possible to generally recommend one over the other (except perhaps "limit WIP" that is quite universal).

- Limit WIP.
  Universally useful to get things done and focus energies on what really matters.
- Break down large work items into smaller ones.
  Perhaps as a policy in a DoD for some of the "preparation" step.
- Swarm.
  The whole team tries to work intensively on one work items to get it done as soon as possible
- Spread knowledge.
  Distribute knowledge so that all team members can potentially work on all work items. Perhaps using a policy dictating that team members lacking knowledge to complete a specific type of work item pull that item anyway; with support or coaching from the team member that has the knowledge.
- Remove "internal blocks".
  Help each other to remove "internal" blocks (team member waiting for another team member).
- Minimize the chances for a work item to become "blocked".
  Perhaps using a Definition of Ready that stipulates specific conditions. For example: if your work items are often blocked due to lack of business expert needed to validate the work done, then you could stipulate that work cannot be started if a given business expert isn't available next week (if such work items usually take one week to complete).
- Increase acceptance of done work items by downstream.
  Perhaps by creating a DoD policy with the help of the down-stream partner to maximize the chances for work items to be accepted asap.
- Focus on high quality.
  Focusing on delivering with higher quality can initially slow down flow (there is perhaps more check), but it will increase flow on the long term as you get rid of a lot of failure demand and re-work (less bugs/expedites)
- Only do the "right" thing.
  Get rid of your huge prioritized backlog, instead use the spice girls questions with the customer when you need to replenish the "to do" queue: "tell me what you want, what you really, really want!"

BOOST 6: Manage the Demand

	Impact	Can it be skipped?
BOOST 6	Outcome: The demand on the team is managed following explicit policies in a sustainable and repeatable way. Output: A ”Ready” queue replenishment policy.	Skip this if the team already has a ”Ready” replenishment policy that actually works. *This feature is required for evolving the team's visualization board and policies into a kanban system.*

You help the team to create policies to replenish the ”Ready” queue.
The team creates policies for the management of the ”Ready” column (prioritization of backlog items, replenishment rules) – and visualize them. This may be quite complex when a team handles several customers or systems.

Prerequisites

The team has completed the Kick-start workshop.

Manage daily Demand

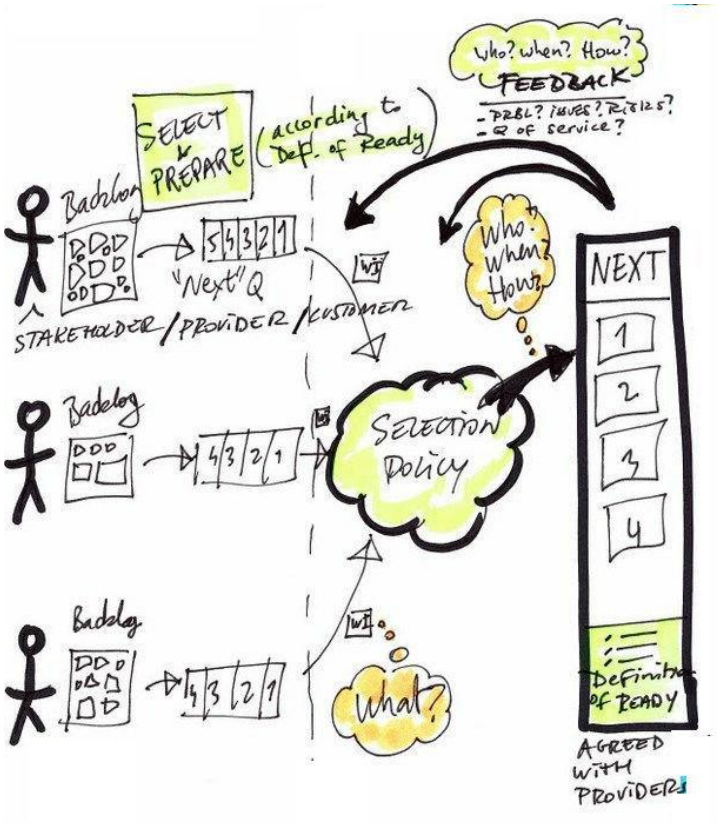
The “Incoming” buffer
If the team manages its own demand, here is a simple policy that can help: use an “Incoming” buffer. Create an “Incoming” buffer to the left of the “Ready” buffer. Team members should use the buffer to place potential new work items that are new since the last planning meeting. At the start of the next planning meeting, the team decides how to handle each new request. The request can be:

- pulled in directly (because it is an expedite or the due date is at risk),
- placed on the ready queue (it is somehow at least as valuable as the other X items in the queue),
- placed in the ‘backlog’ for later, or
- simply discarded?

Explicit policies for managing demand from multiple upstream partners, or systems

Teams that live in complex environments (multiple upstream partners, partners with very different profiles, multiple systems, etc.) may need to set in place a set of explicit policies to be able to treat their demand consistently. For example, teams serving several partners may find it hard to refill the “ready” queue in a way that is *balanced and fair f*or all the customers. Most teams do not have explicit replenishment policies, meaning that the loudest customer usually get more work items in the “ready” queue than the others. That is OK for a while, until other customers complain and the team must explain *why* they have not served them. Which they cannot.

Selection Policies
The team can use an explicit selection policy (algorithm) to replenish the “ready” queue. The selection policy dictates how and when the team pulls the top of each partner's backlog into the “ready” queue. The policy should be made available to all partners so that they understand what service to expect.



Examples of simple policies are:

- *FIFO* (First In First Out): the team select work items in the order they arrive (e.g. ticket creation date) regardless of who created them.
- *Round Robin*: pick the first top work item for each partner in turn.
- *Weighted Round Robin*: pick the first X items for partner P, then pick Z items for partner P+1, go round.

There are of course numerous other possible policies, but remember that the simplest selection policies are the best. When in doubt, try to explain the selection policy to a partner. If it takes more than two sentences, you are probably too complex. If you feel adventurous, check scheduling algorithms and networking algorithm for inspiration.

Discuss with the team what the selection policy should be.
When unsure, pick the simplest policy the team can think of and start to apply it. The partners will surely want to modify or adapt the policy once they start to understand how it affects them.

BOOST 7: Assess Depth of Kanban

	Impact	Can it be skipped?
BOOST 7	Outcome: The team has 1) a better understanding of Kanban, 2) an idea about their current condition, and 3) some input about what to improve to improve their kanban system. Output: An evaluation of the team on the depth of Kanban chart.	You should wait "a while" after a Kick-start, until the team has matured, before running this feature. *This feature is required for improving the team's kanban system.*

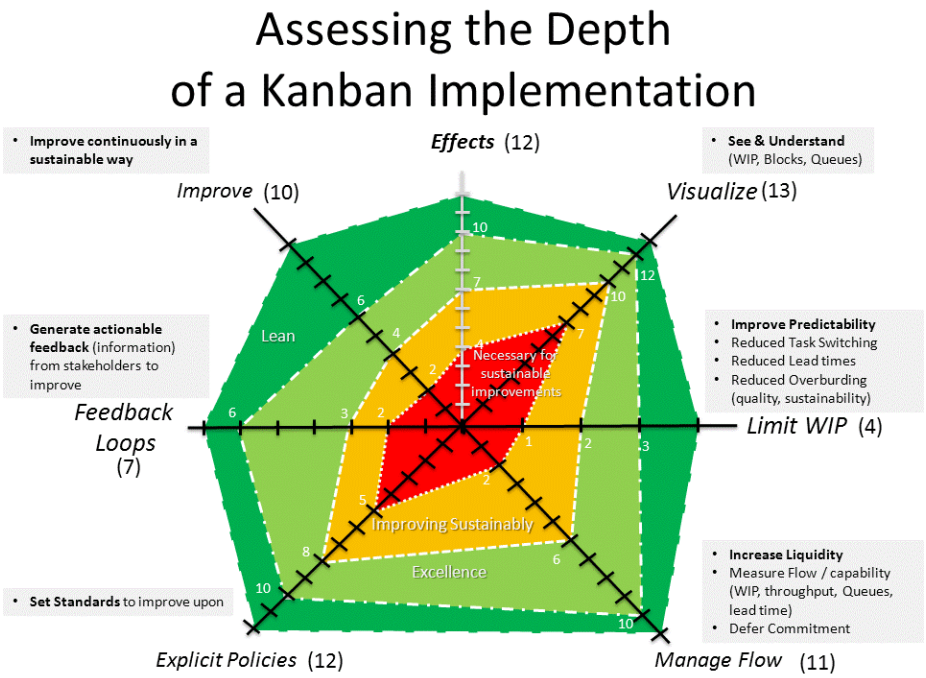
You help the team assess the "depth" of its kanban system and give them input on what can be improved.

Prerequisites

The team has completed the Kick-start workshop and the kanban system has stabilized (you have run BOOST 2 with the team at least once, the team holds its meetings and policies).

Explain Kanban using its 6 Properties

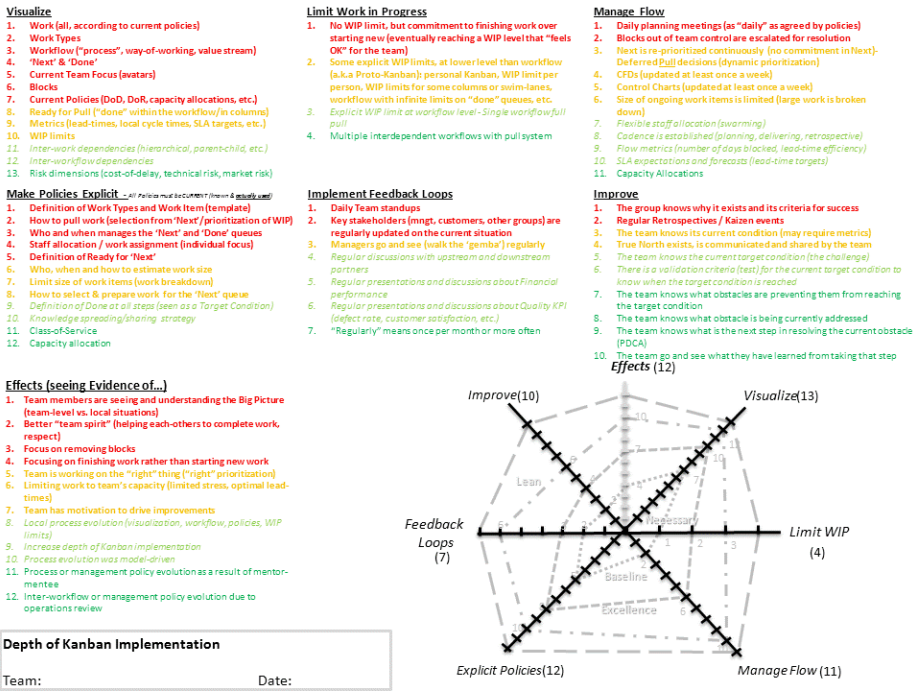
Explain the Kanban method to the team using the 6 properties of Kanban.
The team has used Kanban for a while and can now relate to the properties. This will increase the team's understanding about Kanban and their own situation.
You can get the slides from here: http://www.slideshare.net/ChrisAch/depth-of-a-kanban-implementation



Assess the Team's Kanban Depth

Evaluated how "deep" the team follows the 6 properties.

- Use the following template to check what practices and policies the team is currently using under each area. Missing practices should trigger improvement ideas for the team.
- Visualize the result of the evaluation on the "spider" graph. Help the team to prioritize improvement ideas by identifying the area that needs most improvement.



FAQ

What are these different colors used for?
The primary goal for the use of Kanban at Sandvik IT is to engage the teams to improve, continuously and in a sustainable way. Now, in order to start improving a team must have some minimum capability to see and understand what is going on (clarity of execution) as well as knowing what to aim for (clarity of purpose). The red area on the graph is an attempt to define the minimal depth a team must reach in order to start improving on its own. While the team is "in the red" it cannot improve. That is a clear signal to the coach that action is need a.s.a.p. The other colors indicate other "levels" of depth, the greener the better, called "Improving Sustainably" (you want to team to be there a.s.a.p.), "Excellence" and "Lean". Note that the Kanban Kick-start aims at getting the team directly in the "Improving Sustainably" area. Though, depending on the context, some teams might not manage maintain that condition and fall into the red.

So rather than a sequential implementation of Kanban (visualize, then limit WIP, etc. etc.), it might be better to implement all the reds, then all the yellows, then all the greens?
It depends what you are after when implementing a kanban system. Teams that need to get in control of their situation need to focus on visualization, teams that have worked "ad hoc" for ages probably need to focus most on policies first, while others drowning in WIP must - of course - limit WIP. So, you current condition dictates how you will grow your kanban system. What we are after is to give the team the capability to improve/evolve on their own. That is where the "red" area comes from: a team needs to, at least, have these attributes to be able to understand their current condition and evolve from there. Then, the other colored areas are simply here as a reminder to the team: does it makes sense for us to go further with one principle when we neglected others? As the cost of going "deeper" is probably higher than the cost of addressing "surface" principles, and as the principles affect each-others, the RoI is probably better when addressing all "orange" first, then all "light green", etc. So, yes, we are advocating a "spiral" model to implementing Kanban rather than a linear model.

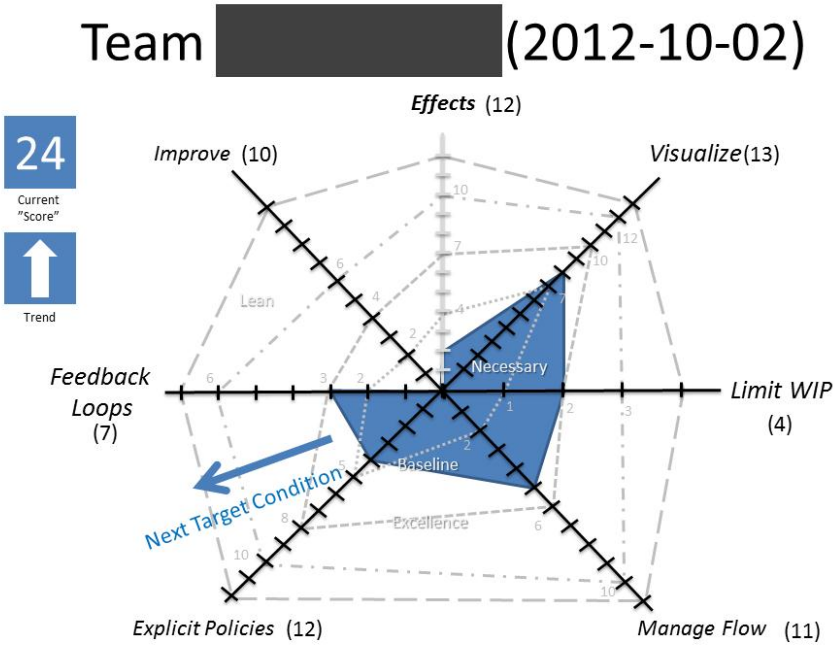
Why do the axes have different scales?
Some principles are more demanding than others and it should be reflected on the graph.

Isn't there too much detail?
Probably! This is the very first version and it has "everything" in it. It is a coaching tool and the coach is needed to moderate the discussion and explain some of the concepts. So, currently this tool is not made for the teams to evaluate themselves. We will continue to use the tool and improve it (simplify it).

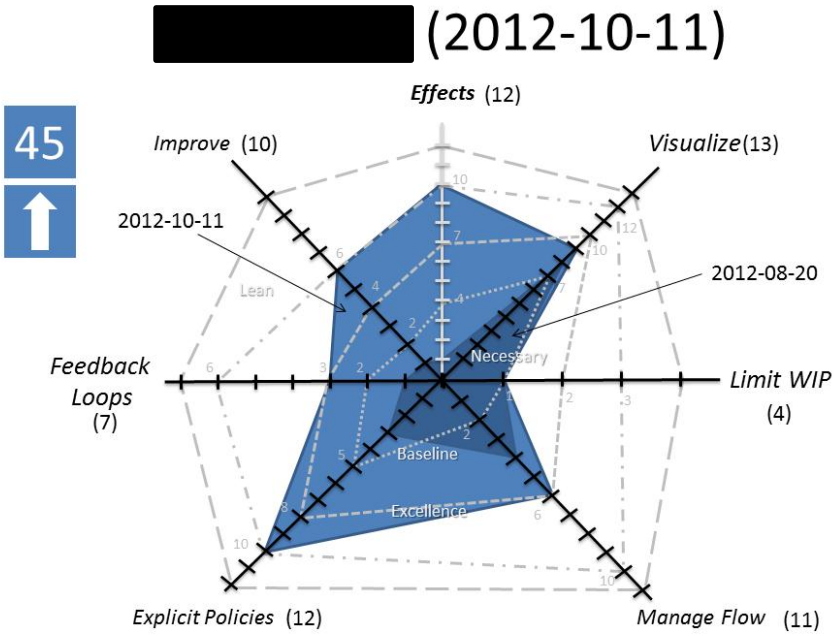
Are these Toyota Kata's 5 questions I see there?
Yes! We have renamed David's *Improve* category to *Effects* (seeing evidenced of...) and used Improve for the improvement kata. We felt that it better matched the approach we have had at Sandvik IT to Kanban as a tool for bootstrapping continuous improvements.

How do you use the graph?
We use the graph in three ways (sometimes all at once):

- *To present what is Kanban* to a team with hands-on experience. Kanban can be presented in many ways. We have found that this description is very useful to a team that has used Kanban for a while, as it is too abstract and too concrete at the same time for green teams.
- *To find out the depth of a Kanban implementation*. We usually sit down with the team lead or the whole team and go through the questions. It is important for the coach to moderate the answers from the team to get constructive answers. For example: Team "Yeah, we definitely get great feedback from the customer!" Coach "In what form and how often?" Team "Every 6 months the customer answer a question 'How pleased are you with the team (graded 1 - 5)!'". The coach may in this case answer 'no' for the team (even if the team thinks it is a 'yes') and explain the rationale behind.
- *To inspire a team to improve*. The "depth" on its own is actually not that interesting, what is really interesting is: what do we do about it? The team may get some improvement ideas simply by going through the questions. But, you get can in situations where the team is really inspired to improve their visualization when they do not even limit WIP. So, to help the team evolve in the right direction, you could use the graph to bring balance to the team's kanban system by setting target conditions related to areas that are left "behind". Here are some examples from two teams:



Team that needs to get better by creating more feedback loops and makes policies explicit.



Team that improved dramatically over 2 months by using the improvement kata to get better policies and flow. Now the focus must be on limit WIP.

Wow, cool! Now you can actually compare two teams to set the right salaries based on how much they score! STOP! The goal here is not to compare teams between them. It really is only a coaching tool for each team to understand *the next smartest improvement to do* for having a well-functioning kanban system. Considering the wide variety of team compositions, context, challenges and technologies, it is wrong to use the "depth of Kanban" to rank teams for management comparisons. *The only valid usage is to compare a team with itself over a period of time*. Did we get better? If no, why? If yes, why?

BOOST 8: Set Next Target Condition

	Impact	Can it be skipped?
BOOST 8	Outcome: The team knows what to problem/issue to resolve in order to improve their capability, in line with their "true north". Output: A target condition.	Run this when the team has become comfortable using its kanban system (following all current policies, new policies are made explicit, planning meetings work). *This feature is the beginning of the next phase: improve relentlessly in a sustainable way.*

This Boost is actually out-of-scope for this Field Guide, as the goal of the Kanban Kick-start is to bring a team to continuous improvement baseline.

Sorry, this boost is still in R&D!
We want this guide to only contain recommendations and How-Tos that have been used many times in the field. Right now, we do not feel that we have sufficiently validated this *boost* to give such recommendations.
In the meanwhile, do you have validated knowledge that you would like to share using this guide? Contact us!

Now that the team understands and can maintain a current condition, you help the team set a target condition to drive improvements.

Prerequisites

The team has completed the Kick-start workshop and the kanban system has stabilized (you have run BOOST 2: Follow-up with the team at least once, the team holds its meetings and policies).
The team has a good understanding of its *current condition*:

- The team visualizes its workflow, WIP, "Ready" and policies
- Explicit Policies are actually used
- The team measures its workflow and knows (at least) the lead-times and throughput for different work types of different sizes.

Identify a Direction for Improvements

Identify a *True North* for the company or unit.
Using the knowledge the team has over its current condition, identify a *Challenge* with the team.

Identify a Target Condition

Identify a *problem* that needs to be resolved for the team to come closer to the completing the challenge.

Understand what is in the way

You may use the "Depth of Kanban" to get hints and ideas on how to solve some of the problems.

Remove obstacles using experiments

Experiment with one potential solution: use your solution for 'a while' and see if your current condition changes in the right direction. If not, pick another potential solution.

BOOST 9: Give Feedback on the Field Guide

	Impact	Can it be skipped?
BOOST 9	Outcome: The field guide gets better and others can benefit from your insights. Output: Some concrete suggestions are sent back to the ones maintaining the field guide.	Run this whenever you have feedback to give on this guide.

Now that you have used this guide, please give feedback to improve it.

Prerequisites

You have read this guide, perhaps even used it in real-life situations.

Just do it

This field guide is not perfect, nor does it have the ambition to be.
Though, it should be easy to use “in the field” and the content should be aligned with the current best known practices in Kanban and continuous improvement. You are more than welcome to give us feedback if you have any suggestion, or complaint, that makes this guide better.

Contact **Christophe Achouiantz**
Twitter: @ChrisAch Blog: http://leanagileprojects.blogspot.se/
christophe.achouiantz@sogeti.se or christophe.achouiantz@gmail.com

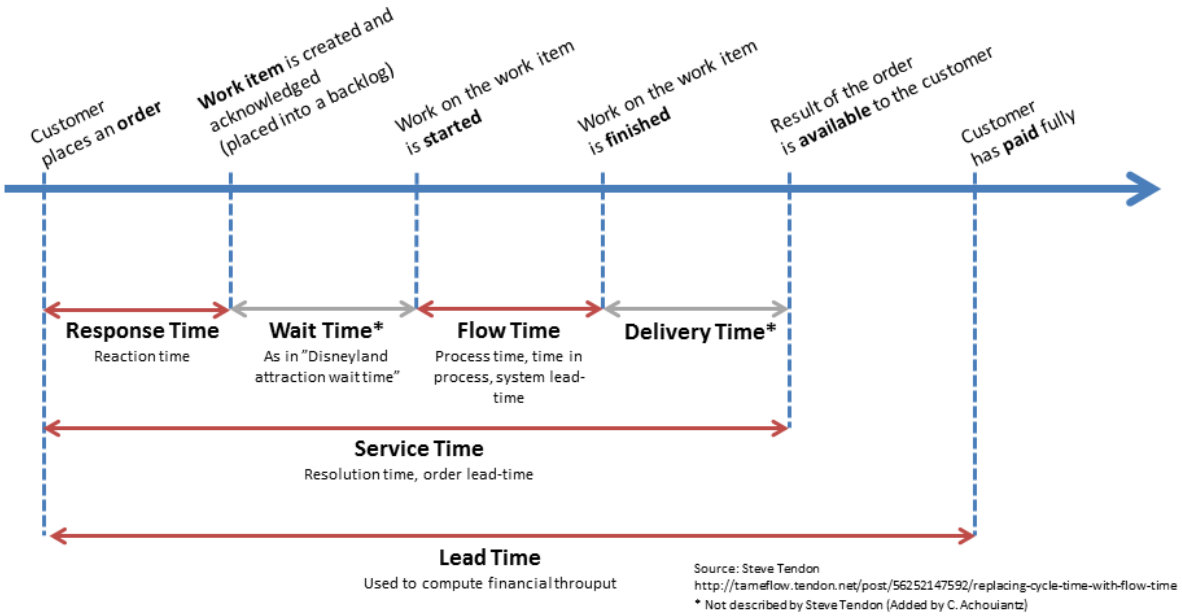
Contact **Johan Nordin**
Twitter: @JohanNordin
johan@simplifychange.se

Thanks!

Appendixes

Appendix I: Definition of Lead-Times

When discussing the capability of a team or of a value stream we often use different lead-times. Here is a simple and elegant definition of different lead-times that are of interest, these definitions are the work of Steve Tendon (http://tameflow.tendon.net).



Definition of different lead-times (Steve Tendon)

- **Flow Time (FT)**, or "Process Time" or "Time in Process" (TIP): The time from the moment work is started to the moment work is finished; in other words the time that a work item spends as WIP (Work in Process). Flow time is used to compute throughput in terms of units of productivity. This is sometimes known as Process Time. We will use Flow Time to compute production throughput or operational throughput and to exploit Little's Law. Note that in the Kanban Community, flow time is often referred to as lead time (and in particular as "system lead time"); though we will use lead time in a different sense as explained below. Other synonyms for this concept are: cycle time, throughput time, and sojourn time; though they might be ambiguous.
- **Lead Time (LT)**: The time from the moment a customer makes an order to the moment the customer has fully paid. We use LT to compute financial throughput performance (in the throughput accounting sense of TOC, i.e. money per period of time). Note that this is different from the that used in conventional business, where it relates to the period between order and delivery. We could refer to the latter as "Order Lead Time" signifying the amount of time an order has to be placed ahead of when its delivery is needed; but, as we will see below, we prefer to use Service Time instead.
- **Response Time (RT)**, or "Reaction Time": The time between the moment a customer makes an order and the moment when that order is actually acknowledged as a work item that will be handled by the service organization or team. In an Agile/Scrum setting, this would be the moment that the work item is added to the "backlog."
- **Service Time (ST)**, or "Resolution Time": The time from the moment a customer makes an order to the moment work is delivered back to the customer. This is often known as the (Order) Lead Time. It is used mainly to compute throughput performance in terms of customer service (even when manufacturing and delivering tangible goods). While (Order) Lead Time might be what is considered when negotiating terms with customers, Service Time puts the emphasis on your own organization's responsibility in servicing a customer need.

Frequently Ask Questions (FAQ)

Must the Kick-start workshop take a whole day? Can't it take half a day?
It all depends on the maturity of the team you are working with and your goals.
A team using Scrum will "get it" much faster than a team working ad-hoc for ages.
The question boils down to how much coaching/support are you prepare to do after the workshop.

Why not limit WIP directly?
The focus of the Kick-start is not necessarily to optimize flow as soon as possible. The goal is to reach a continuous improvement baseline, a sustainable current condition. Therefore, it may be more interesting to make current policies and workflow explicit for the team to *understand* its current condition, rather than directly start optimizing flow.

What's Next?

We are experimenting and gathering knowledge in several areas, but we do not feel that we yet have sufficiently validated knowledge to have it in a field guide. Future versions of this guide, or *other field guides*, may contain:

- **Specific practices and recommendations for teams above tier 1** (i.e. team that use or coordinate other teams). These teams need to see and understand multiple dimensions in order to get a good understanding of the current condition. This may require several boards, or one board with several dimensions. For example: team activities, status of the initiative/project/program/unit, exception handling (risks, opportunities, problems).
- **Operational Reviews**. It is our understanding that operational reviews are needed for improving the whole. We plan to describe how we setup these reviews.
- **Lean/Kanban Guild**. The Kanban users need to share their knowledge and experience using a community of practices. We plan to describe how to setup a guild.
- **Coach the Coach**. Sandvik IT needs more Lean/Kanban coaches to scale-up. We plan to describe how we do it.
- **Improvement Kata and coaching Kata.** These are central to continuous improvements. Yet, we do not have enough validated knowledge with the Toyota Kata to come up with how-tos and recommendations.
- **Leadership standard**. Managers need to go and see, support and challenge the teams. We are just starting our journey in this area.

References

"Kanban: Successful Evolutionary Change for Your Technology Business" David J. Anderson (2010)
"Toyota Kata" by Mike Rother (2009).
"Agile Retrospectives: Making Good Teams Great" Esther Derby, Diana Larsen (2006)

About the Kanban Kick-start Field Guide

The Kanban Kick-start Field Guide is a practical guide to help you kick-start your team into using the Kanban method. It concentrates the experience from introducing Kanban to 50+ teams at Sandvik IT.

Authors

Christophe Achouiantz
Christophe Achouiantz is a Lean/Agile consultant for Sogeti with a passion for discovering new ways to improve continuously. For the past six years, Christophe has been helping software development companies to timely deliver the right thing, at the right quality and the right cost. Before becoming a consultant, Christophe worked as development manager in a start-up where for 7 years he learned - in a very concrete way - the lean and agile principles he advocates today.

Twitter: @ChrisAch *Blog*: http://leanagileprojects.blogspot.se/
LinkedIn: se.linkedin.com/pub/christophe-achouiantz/0/800/a6
E-mail: christophe.achouiantz@sogeti.se or christophe.achouiantz@gmail.com

Johan Nordin
Johan is a Lean coach with a burning interest in learning - by doing - how to effectively inject a continuous improvement capability into an organization's DNA. As Development Support Manager at Sandvik IT between 2008 and 2013, Johan was responsible for creating strategies and concepts that would create a sustainable improvement baseline and desire to evolve towards Operational Excellence. A work that resulted in helping more than 50 Kanban teams.

Twitter: @JohanNordin *LinkedIn:* se.linkedin.com/in/johanhnordin
E-mail: johan@simplifychange.se

Sandvik IT
Sandvik, founded in 1862, is a high-technology, engineering group. The Sandvik Group conducts operations in five business areas: mining, machining solutions, materials technology, construction and venture. Worldwide business activities are conducted through representation in more than 130 countries. In 2012 the Group had about 49,000 employees with annual sales of approximately 11,880 MEUR. Sandvik IT is supporting the Sandvik group's IT needs. In 2012, Sandvik IT has about 900 employees in 40 countries, supporting 365 applications for more than 30.000 users.

http://www.sandvik.com
Facebook: https://www.facebook.com/sandvik