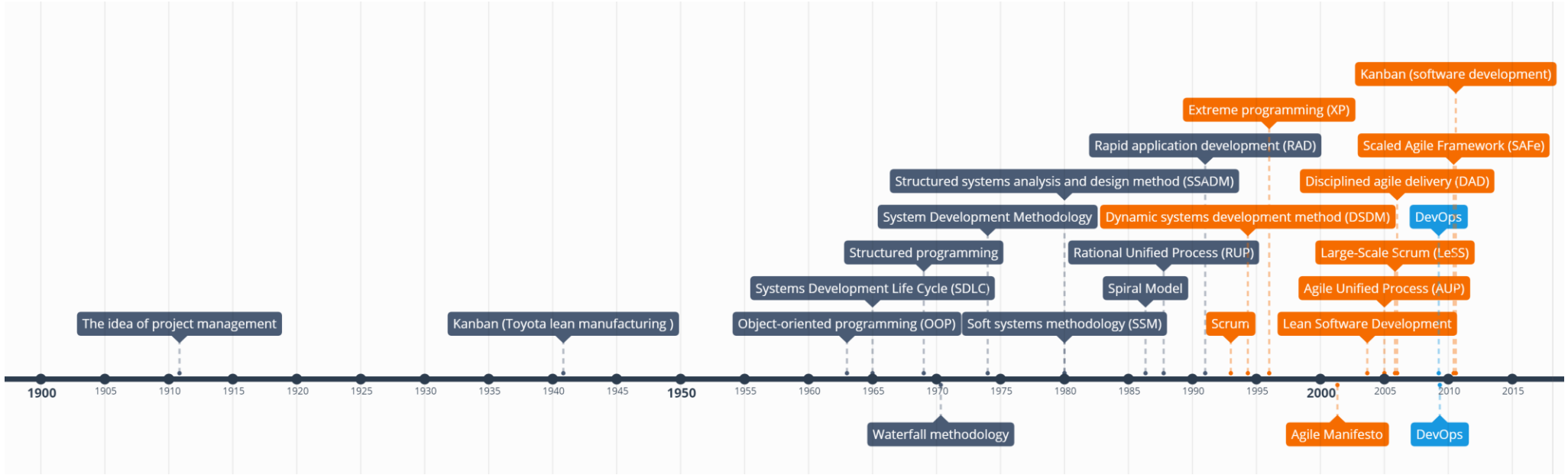
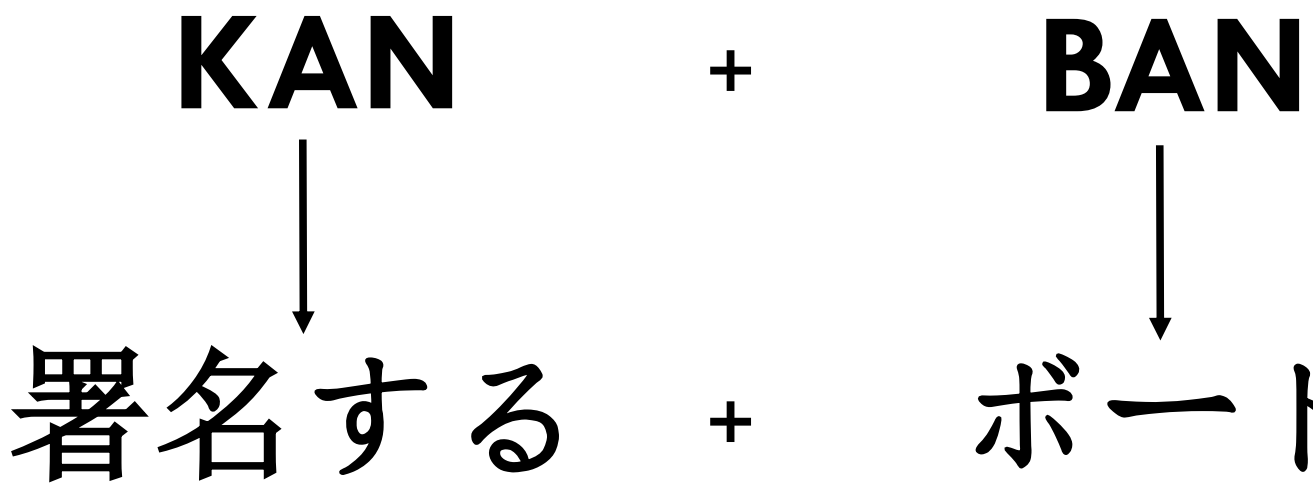
AGILE METHODOLOGIES: KANBAN

Lecturer:
Ehsan Shoja

SOFTWARE DEVELOPMENT PROCESS HISTORY

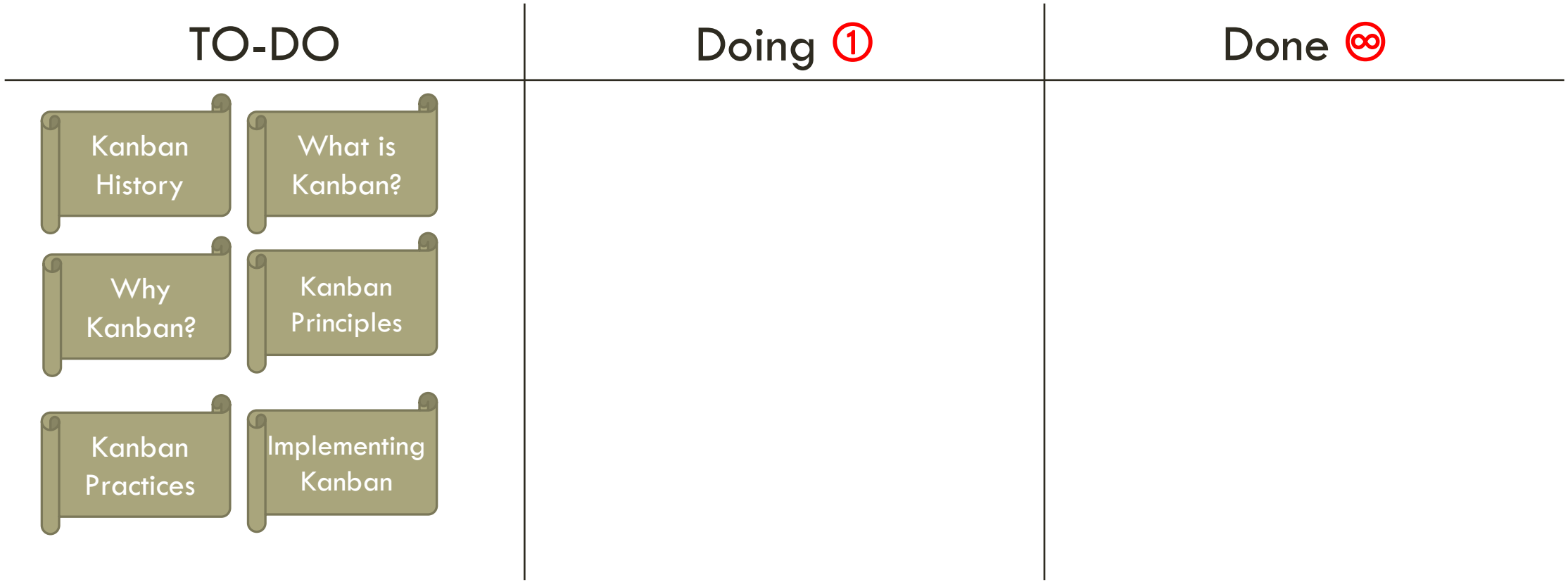




=

“signboard”

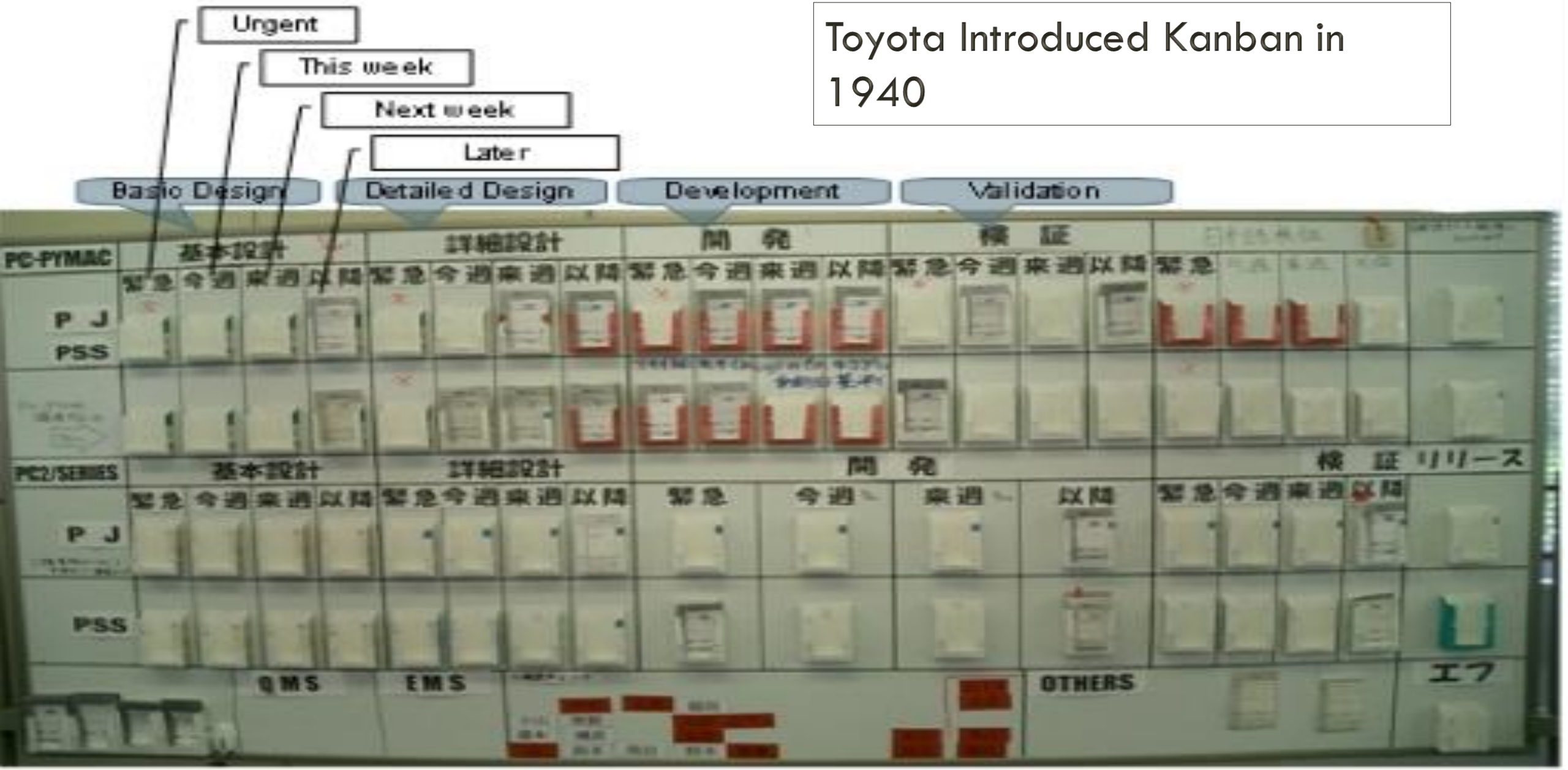
TO-DO LIST


TO-DO
Doing ①
Done ∞
Kanban History
What is Kanban?
Why Kanban?
Kanban Principles
Kanban Practices
Implementing Kanban

KANBAN: HISTORY

- 看板 – Kanban literally means “visual card,” “signboard,” or “billboard.”
- Developed by Taiichi Ohno at Toyota in 1940’s
- Toyota originally used Kanban cards to limit the amount of inventory tied up in “work in progress” on a manufacturing floor
- The Kanban system is a method of using cards as visual signals for triggering or controlling the flow of materials or parts during the production process.

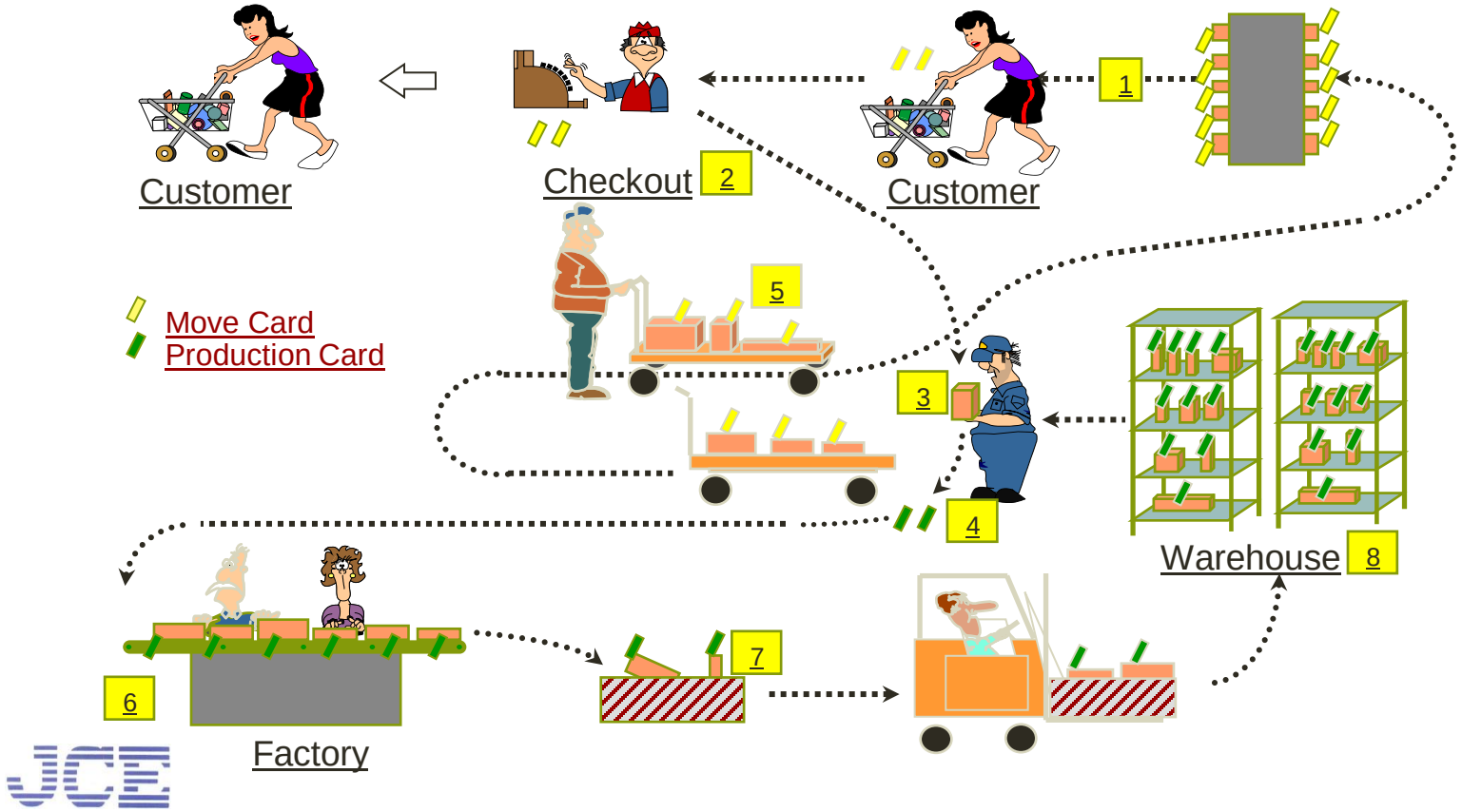
Toyota Introduced Kanban in 1940



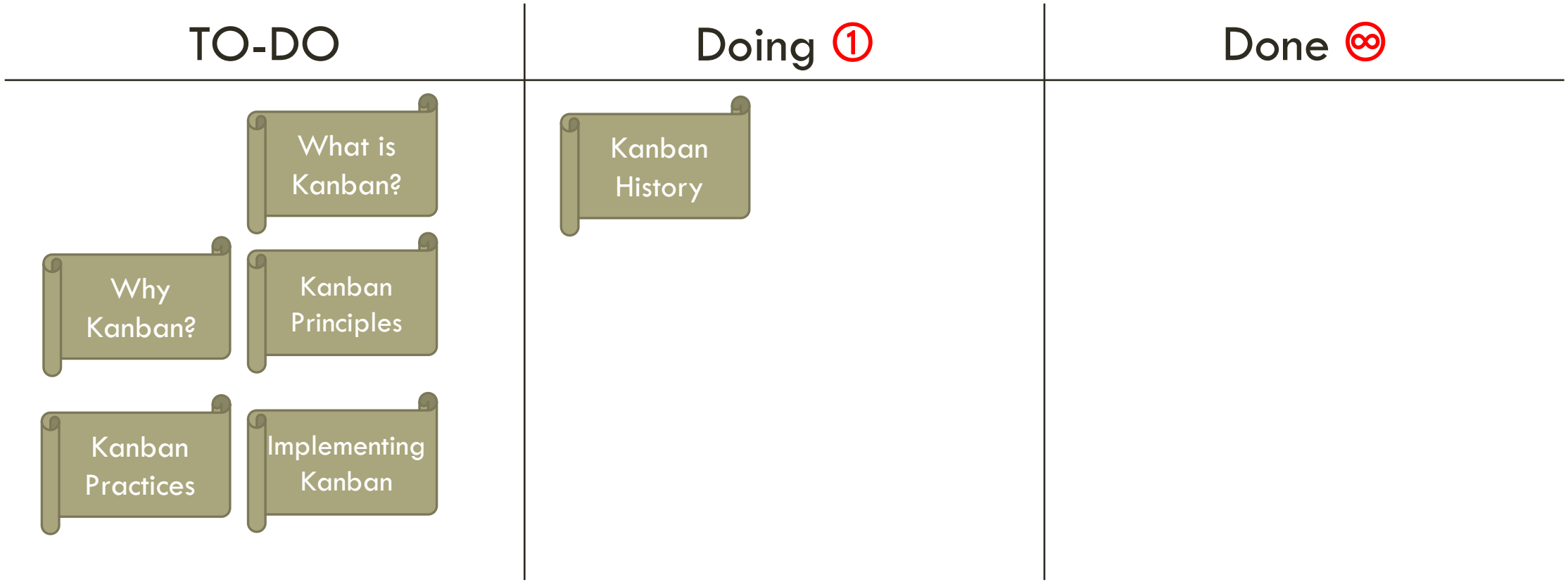
KANBAN: HISTORY

- Designed after the shelf-stocking techniques used by supermarkets
- Demand controlled system where replenishment happened based on market conditions
- Based on a pull based system rather than a push based one
- Kanban not only reduces excess inventory waste, but also the time spent in producing it.
- Kanban cards act as a form of “currency” representing how WIP is allowed in a system.

KANBAN: HISTORY



TO-DO LIST


TO-DO
Doing ①
Done ∞
What is Kanban?
Why Kanban?
Kanban Principles
Kanban Practices
Implementing Kanban
Kanban History

WHAT IS KANBAN?

- Scheduling system used in manufacturing to help companies improve their production process
- "The Kanban Method" for software development pioneered by David J. Anderson
- Adopted for JIT delivery without burdening developers
- WIP limited pull system which exposes system problems through visualization
- In its simplest form, a Kanban system consists of a big board with story cards

WHAT IS KANBAN?

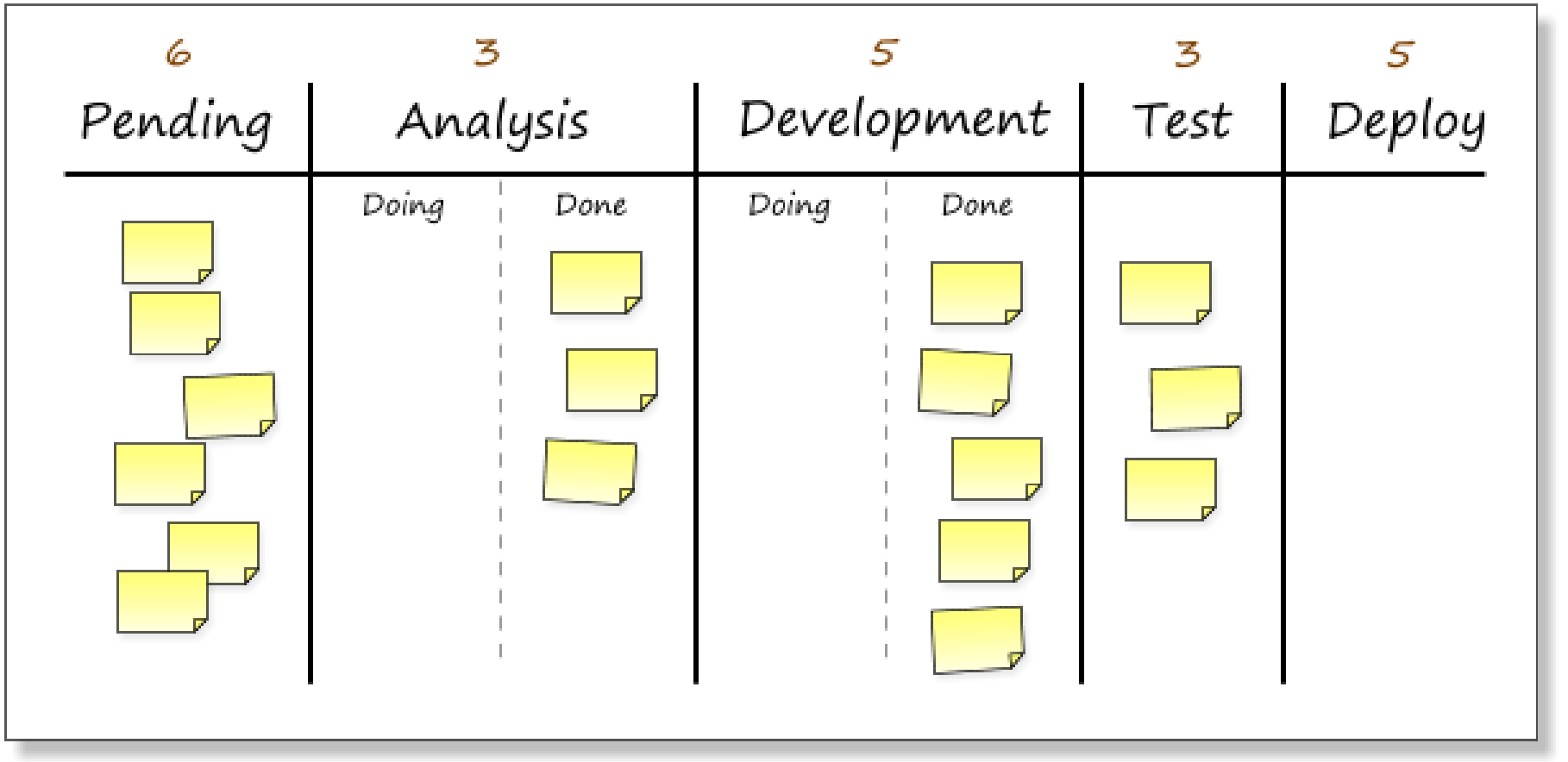
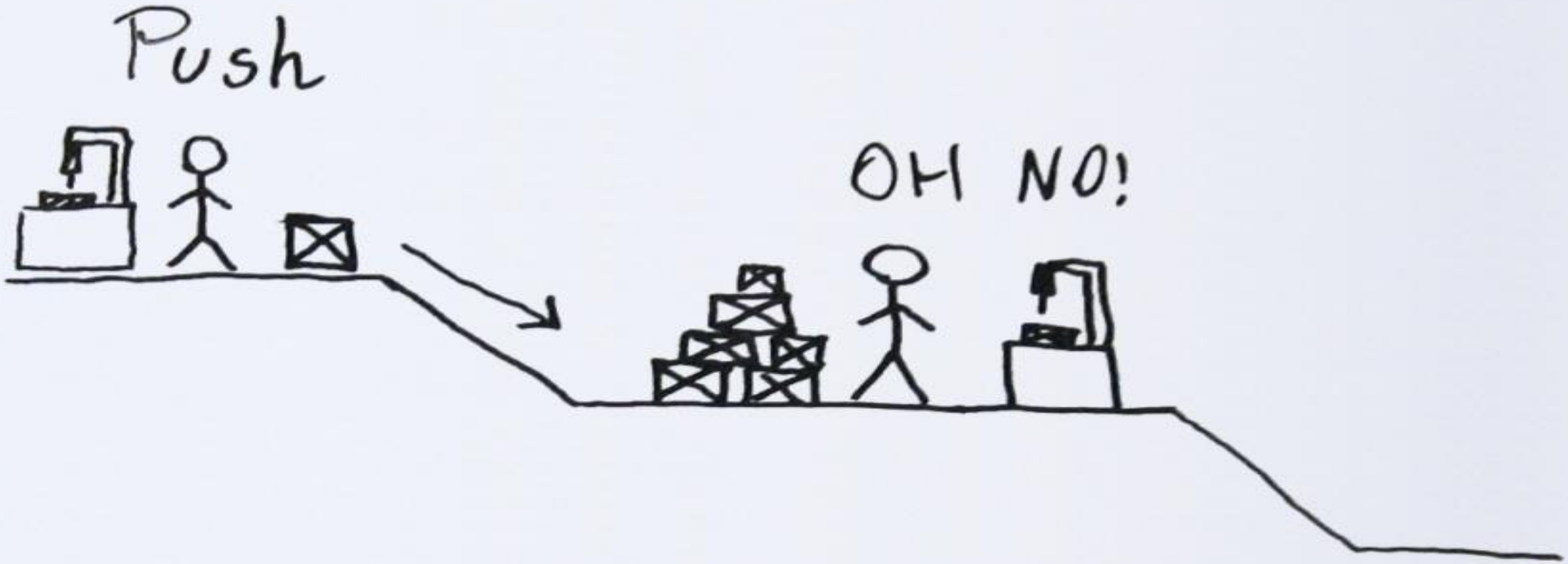


Fig. **One typical Kanban board**

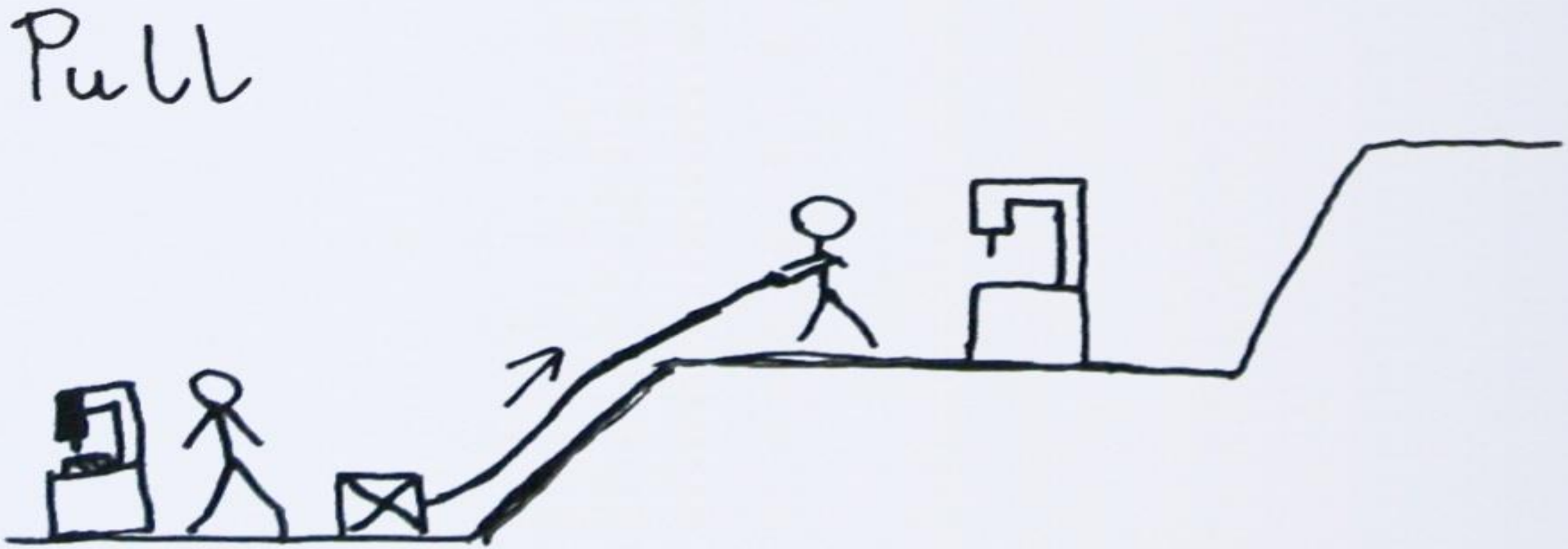
WHAT IS KANBAN?

- Board represents the state of the project at any point
- Different from other visualizations – implements WIP limits
- Tries to limit the amount of work at any stage
- Easy identification of bottlenecks in system through visual boards
- Aims at minimizing waste states

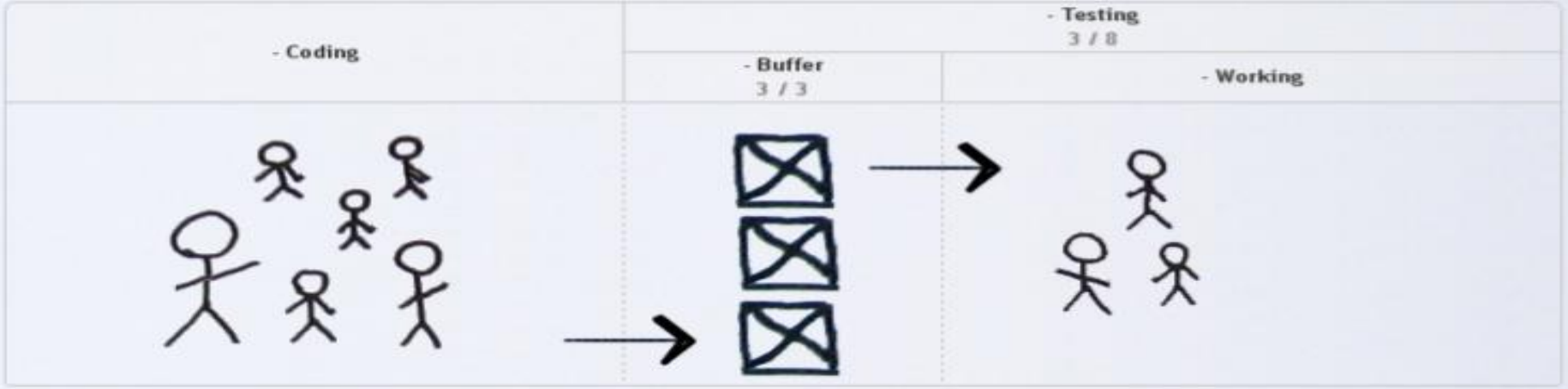
... and pushes more work than another one can actually handle.



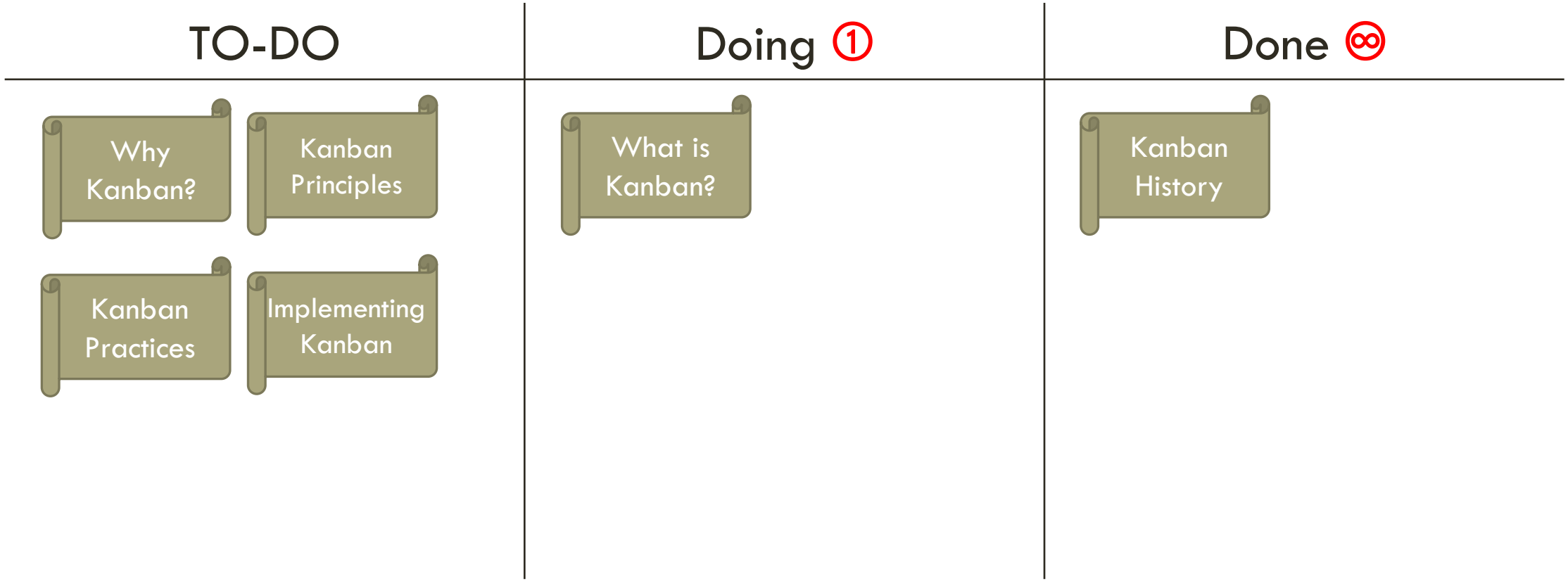
A solution to this is a **pull system**,
where next team pulls work only when ready for it.



You can implement pull system by adding a limited capacity buffer between teams.



TO-DO LIST

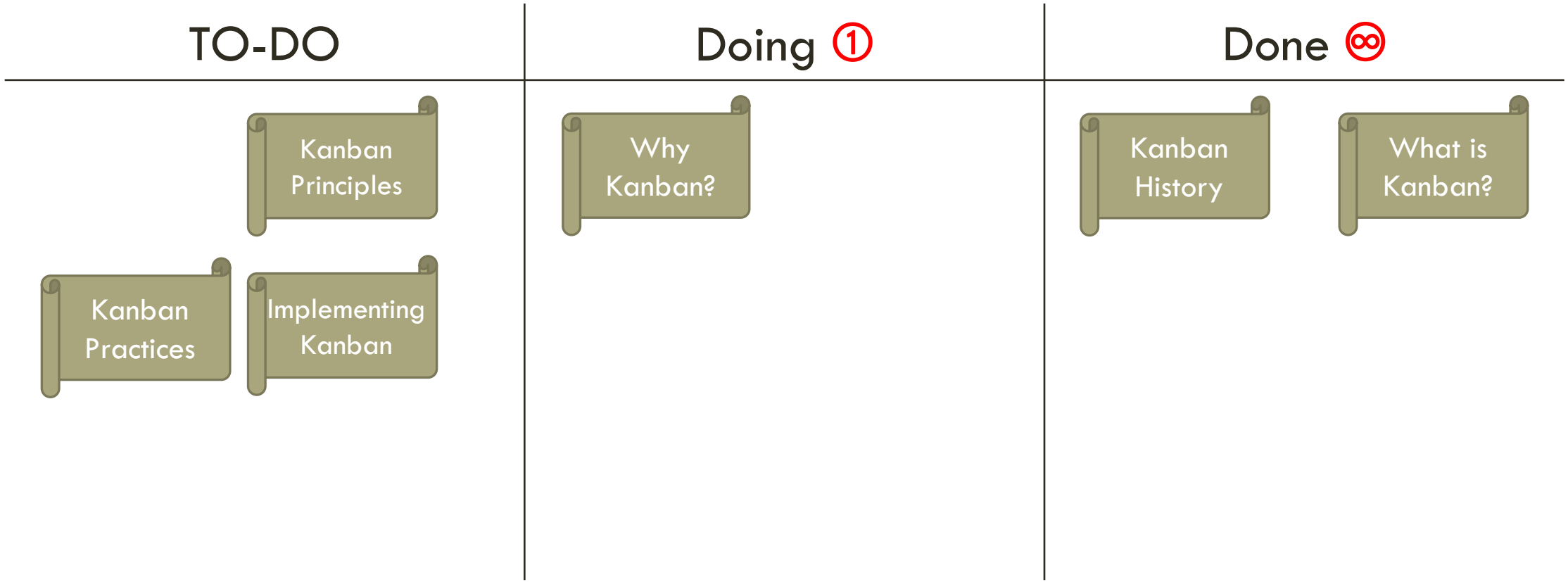

TO-DO
Doing ①
Done ∞
Why Kanban?
Kanban Principles
Kanban Practices
Implementing Kanban
What is Kanban?
Kanban History

WHY KANBAN?

… because …

- it helps in visualizing the system and expose problems
- it allows us to evaluate the impacts of process changes
- it allows us to identify bottlenecks and alleviate them
- it allows us to establish trust in the process
- it helps us to maintain a sustainable pace with a sustainable throughput
- you need to relax and Kanban advocates just that!

TO-DO LIST


TO-DO
Doing ①
Done ∞
Kanban Principles
Kanban Practices
Implementing Kanban
Why Kanban?
Kanban History
What is Kanban?

KANBAN PRINCIPLES

1. Start with what you do now!
2. Agree to pursue incremental, evolutionary change
3. Respect the current process, roles, responsibilities
4. Leadership at all levels

KANBAN PRINCIPLES: START WITH WHAT YOU DO NOW!

- Do not prescribe any new roles or responsibilities to implement the new system
- No such thing as “Kanban Software Development Process”
- Implement Kanban with existing system

- David Anderson

KANBAN PRINCIPLES: AGREE TO PURSUE INCREMENTAL, EVOLUTIONARY CHANGE

- Optimize what already exists
- Agree to continuous, incremental and evolutionary change to improve the system
- Keep experimenting to understand the effects of changes on the system
- Make small changes rather than huge process changes

- David Anderson

KANBAN PRINCIPLES:
RESPECT THE CURRENT PROCESS, ROLES, RESPONSIBILITIES

- Do not remove existing roles and titles
- This will eliminate fears in introducing the new system in the organization
- Will help you get broader support in introducing the new system
- Kanban was designed to reduce resistance to change

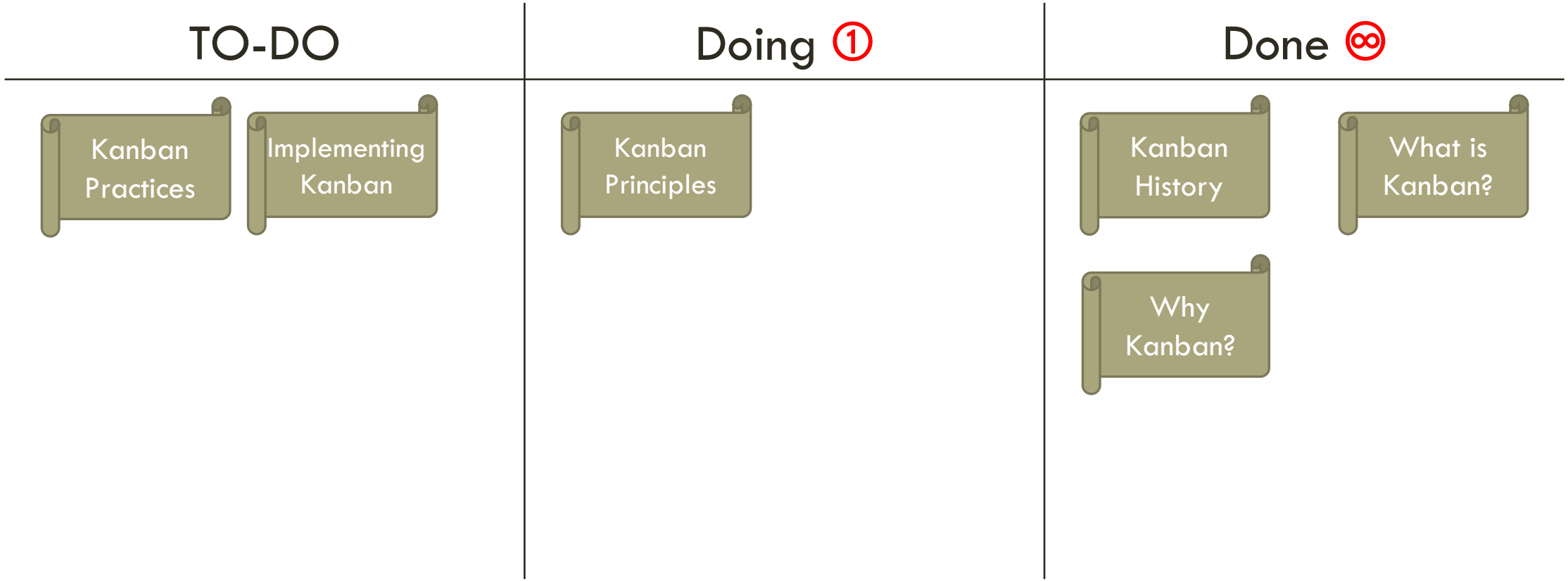
- David Anderson

KANBAN PRINCIPLES: LEADERSHIP AT ALL LEVELS

- Empower the workforce to bring about change
- Swarm on a bottleneck for faster resolution
- Hold frequent discussions and process improvements
- Include everyone in these discussions and do not disregard anyone's viewpoint

- David Anderson

TO-DO LIST


TO-DO
Doing ①
Done ∞
Kanban Practices
Implementing Kanban
Kanban Principles
Kanban History
What is Kanban?
Why Kanban?

KANBAN PRACTICES

1. Visualize
2. Limit Work-In-Progress (WIP)
3. Manage flow
4. Make policies explicit
5. Implement feedback loops
6. Improve collaboratively, evolve experimentally

VISUALIZE (THE WORK, THE WORKFLOW, AND THE BUSINESS RISKS)

- Workflow is inherently invisible
- Visualization is core to Kanban
- Enables people to take a quick look at the state of the workflow
- Use of story cards can be used
- Development process is divided into columns
- Each task is specified on a story card
- Essentially cards move along the board to show workflow.

LIMIT WIP

- Apply limits on WIP in each phase of development
- Is the basis for implementing a pull based system
- Work is pulled into the next phase once capacity is available
- Improves quality by giving greater focus to fewer tasks
- Also reduces lead time for work by reducing the number of concerns for the developer

LIMIT WIP (CONTD.)

- Because maximum utilization of resources is not desirable contrary to popular belief
- Brings in slack into the system – creates a more conducive work process
- Get the most important things done, one by one, with a clear focus
- Things get done faster, better than before, leading to lesser rework

MANAGE FLOW

- Workflow should be closely monitored
- Measurements must be made to identify problems in the system
- Leads to better understanding of the system and helps in making educated improvements
- Helps identify the positive and negative impact of changes introduced in the system

MAKE POLICIES EXPLICIT

- All policies related to workflow management should be explicit
- For eg. WIP limits, basic workflow, rejection/acceptance flow, definition of doneness etc.
- Helps in providing a basis for process improvement based on statistics
- Allows for a more rational approach to process improvement by logical reasoning

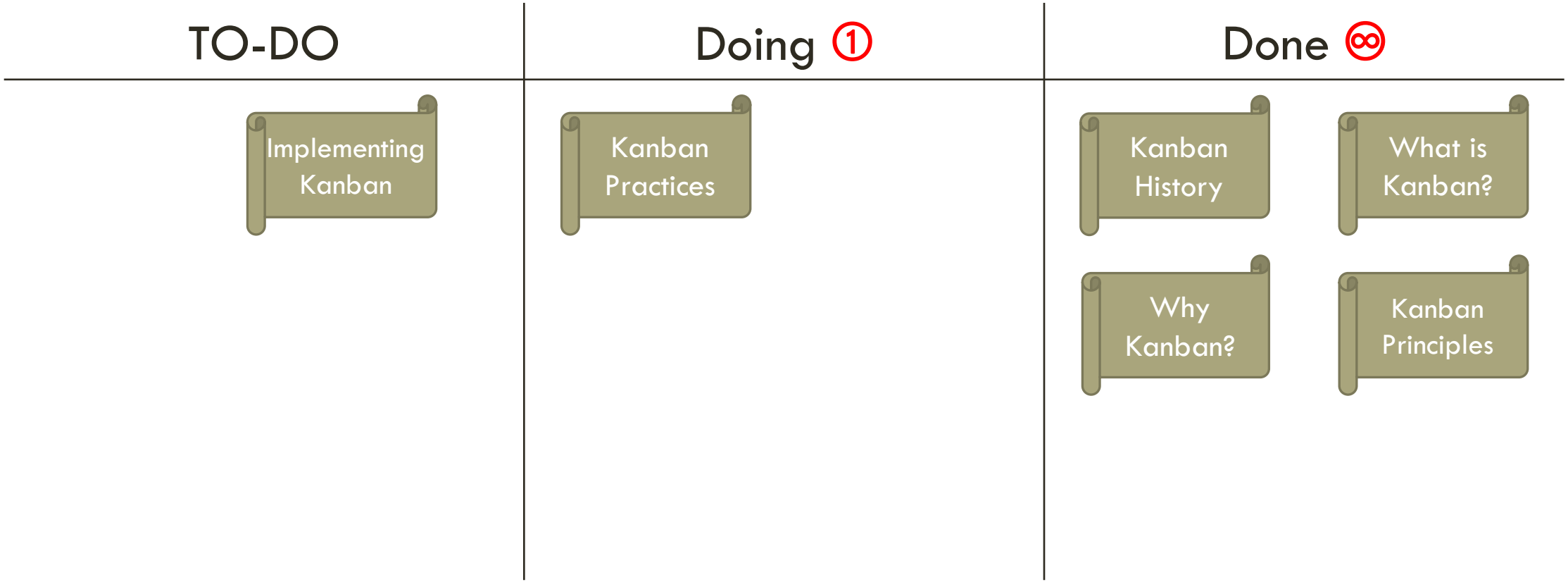
IMPLEMENT FEEDBACK LOOPS

- In Kanban feedback is gathered at different stages of a project: during meetings or at delivery, operational and risk reviews.
- The frequency and format of feedback depend on what has been already established by the Project Office. Any gaps are filled as Kanban identifies them.

IMPROVE COLLABORATIVELY, EVOLVE EXPERIMENTALLY

- Through the use of scientific models
  - The Theory of Constraints
  - The Theory of Profound Knowledge
  - The Lean Economic Model
- Use of models allows a team to make predictions about a change
- The expected and actual result can then be used effectively to improve the process
- This approach leads to learning both at individual and organizational level

TO-DO LIST


TO-DO
Doing ①
Done ∞
Implementing Kanban
Kanban Practices
Kanban History
What is Kanban?
Why Kanban?
Kanban Principles

IMPLEMENTING KANBAN

Things you need:

- A board
- Lots of Post-it notes (preferably of different colors)
- And lots of commitment *(very important)*
- *The next slides!*

AND SOME TERMS...

Important terms:

- *Lead Time* – time taken from request of feature to its completion
- *Cycle Time* – time taken to finish the task
- *Throughput* – essentially refers to productivity. Defined as the amount of work delivered in a time frame
- *WIP Limit Value Stream* – this refers essentially to your development process
- *Swarm(ing)* – collaboration on a problem

THE BOARD

- Allows easy visualization of the development process
- Each column represents one phase in your existing development process
- Numbers on top represent WIP limits
- The number of tasks in each phase is limited by the WIP limits specified

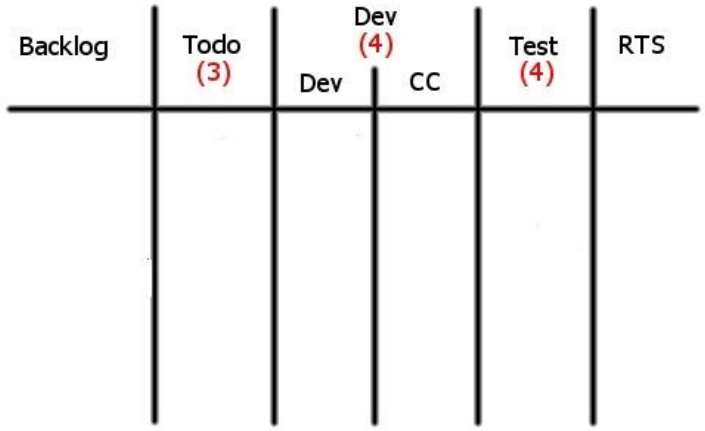


Fig. The Kanban Board

STORY CARD

- Keeps track of features/tasks
- Is more of an XP related feature
- Includes information regarding transition of features on board
- Post-it notes can be used
- Different colored post-it notes can be used for different issue types such as bugs, features, tasks, improvement etc.
- TIP – Token, Inscription, Placement

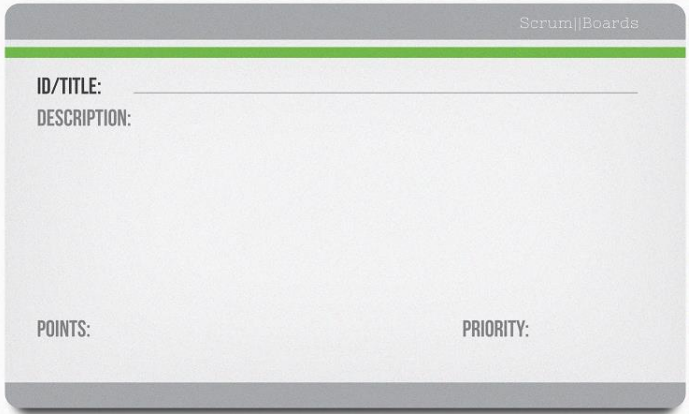


Fig. Story Card

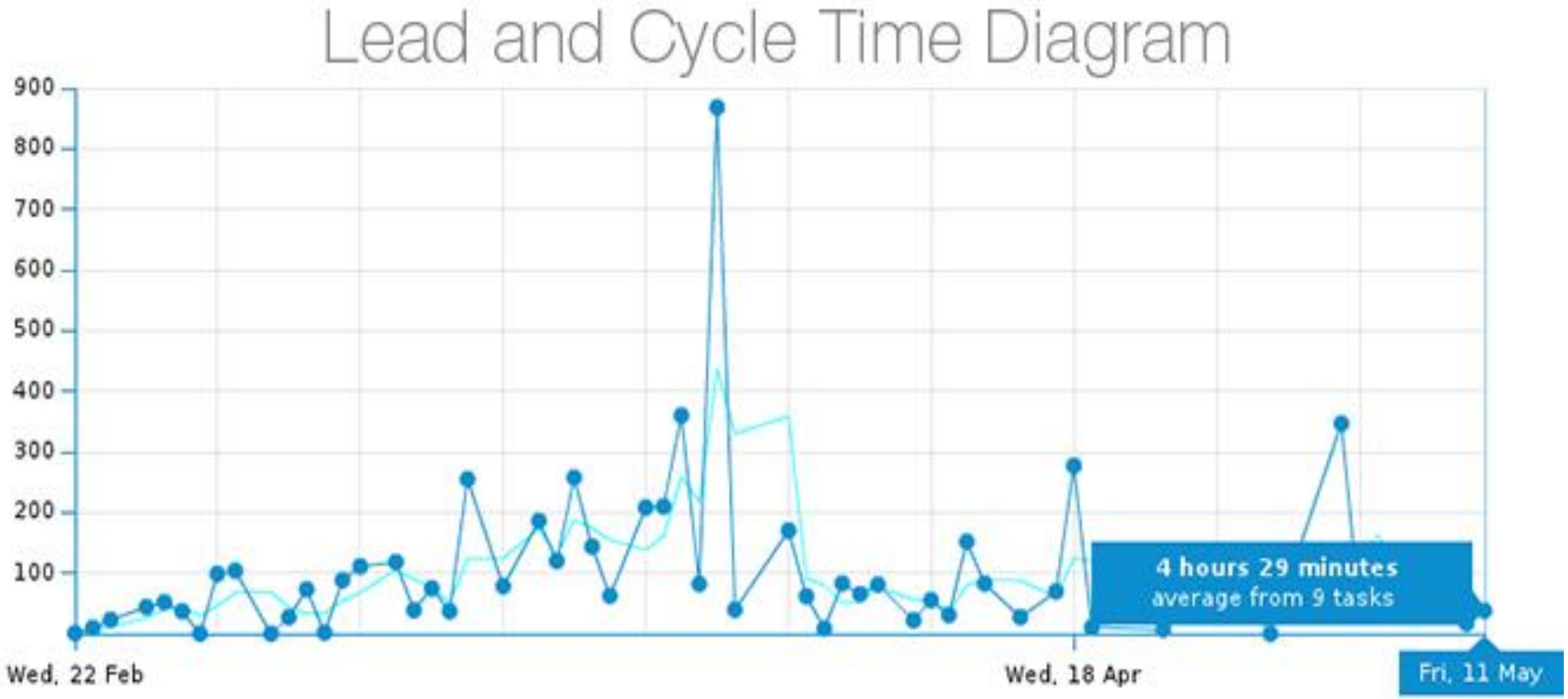
TIP – TOKEN, INSCRIPTION, PLACEMENT (SIDE NOTE)

- Kanban board can be viewed as multi-variant displays, visualizing multiple possible dimensions of a Kanban system (A Kanban Visualization TIP)
- Token: The Token is the element which represents some piece of work, with the attributes **material, size, color** and **shape** all able to represent information.
- Inscription: The Inscription is any detail added to the token about the work. Common types of inscription are **annotations, graphics, linkage** and **formatting**.
- Placement: The Placement is how the Token is positioned on the Kanban board, with the **location, alignment** and **rotation** all being relevant.

CHARTS

- Measurement tools to measure the effectiveness of the system
- Every time card is pushed/pulled on/off the board, charts start changing
- Can be used to interpret various important metrics like average time taken for a task to be completed.
- Can be used to identify the flow of work
- Also useful to identify the state of tasks in each phase of development
- Control Charts & Cumulative Flow Diagrams

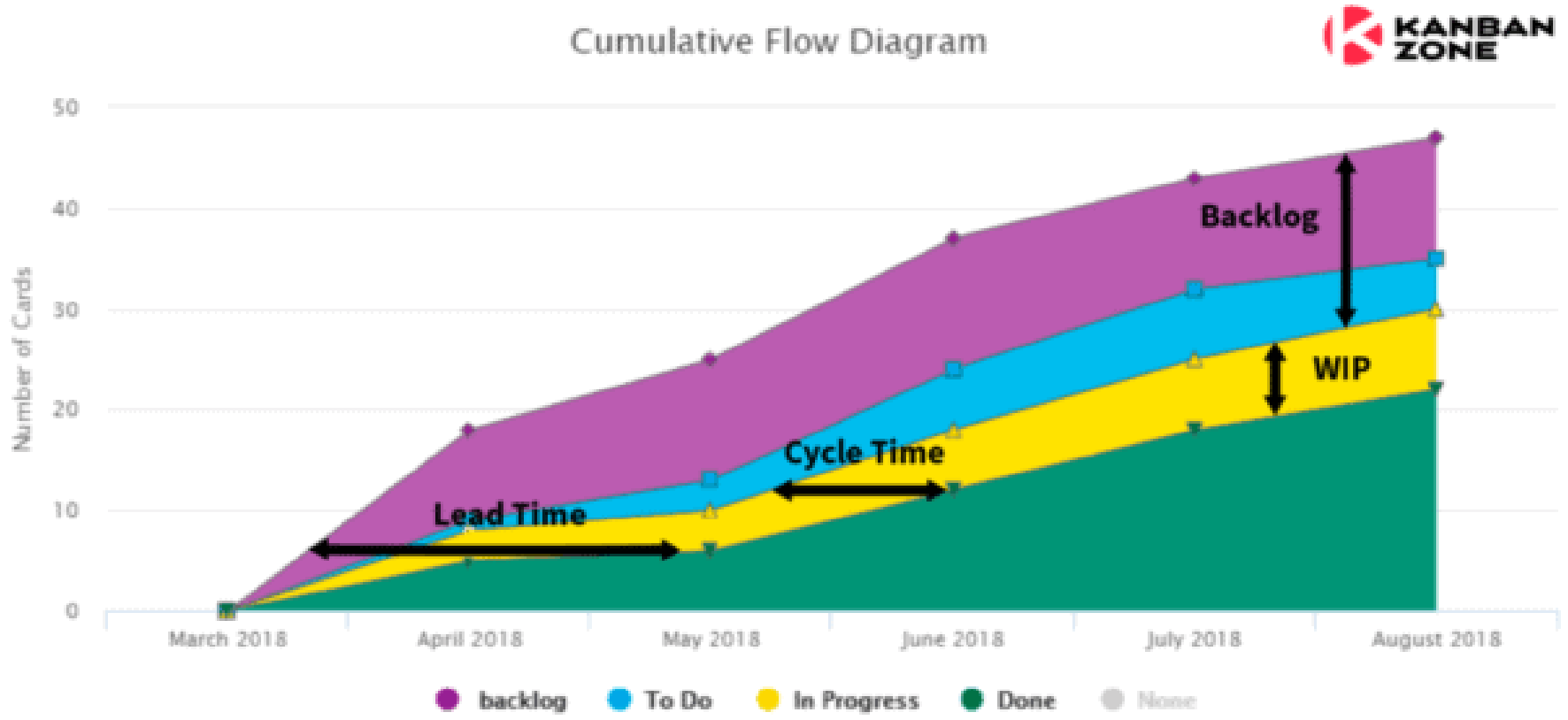
CONTROL CHART



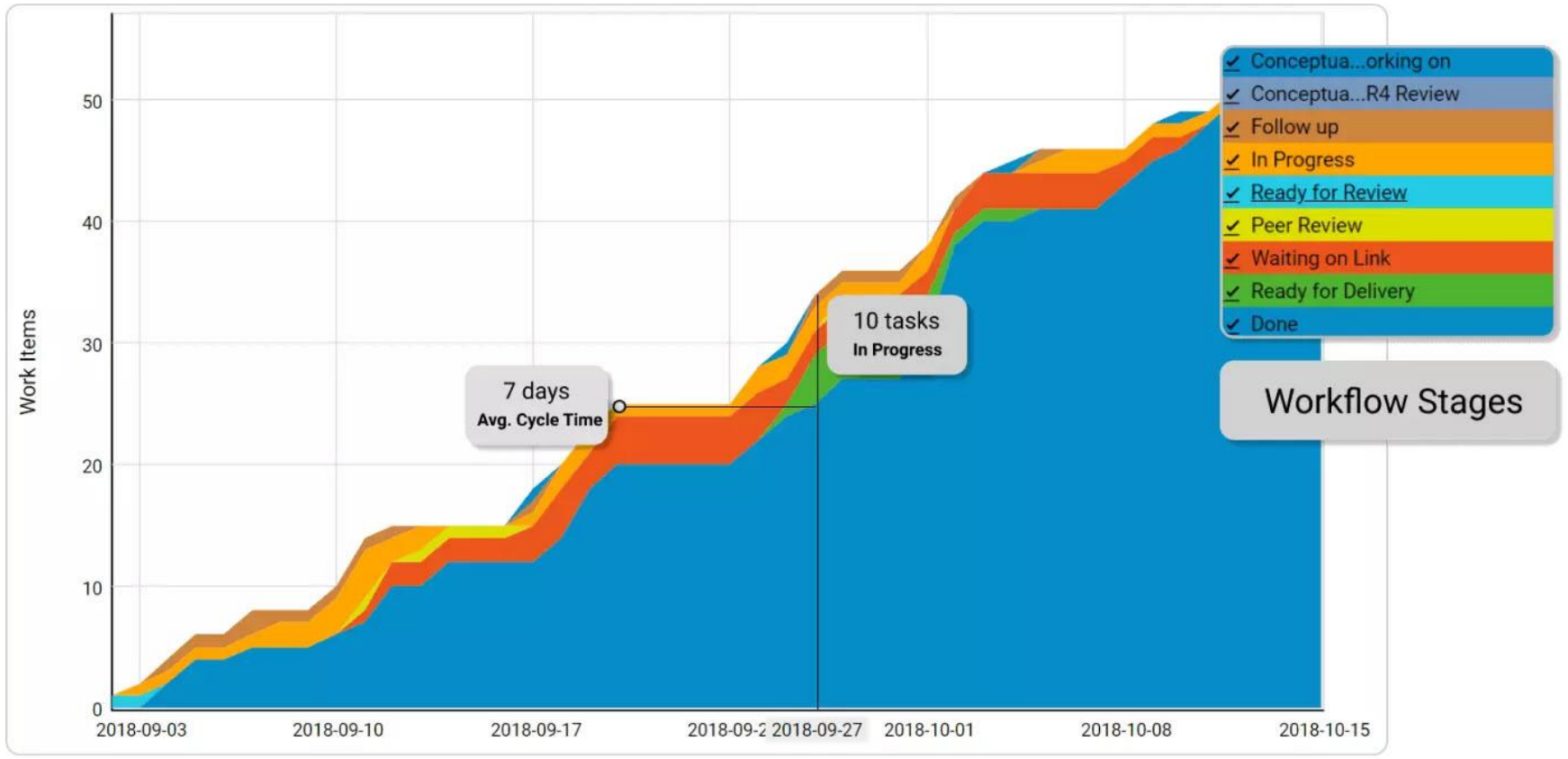
CONTROL CHART

- Are used to measure the average time taken for a task to be processed
- Lead time and cycle time is represented on a control chart
- Simplest charts that can be drawn
- The aim is to keep lead time and cycle time as low as possible

CUMULATIVE FLOW DIAGRAMS



CUMULATIVE FLOW DIAGRAMS



CUMULATIVE FLOW DIAGRAMS

- Show relative amount of work for each stage
- Use of colored areas for each phase for easy identification of bottlenecks
- Vertical distance of the chart shows how many tasks are on the board and helps you set right WIP limits
- Horizontal distance allows you to monitor *Cycle Time*
- CFD should run smoothly
  - Large steps or horizontal lines indicate problems in flow
  - Variations in gap/band indicate bottlenecks
- When the band gets too wide, it indicates problems in work finishing or developers unable to handle amount of work.

LET'S GET STARTED

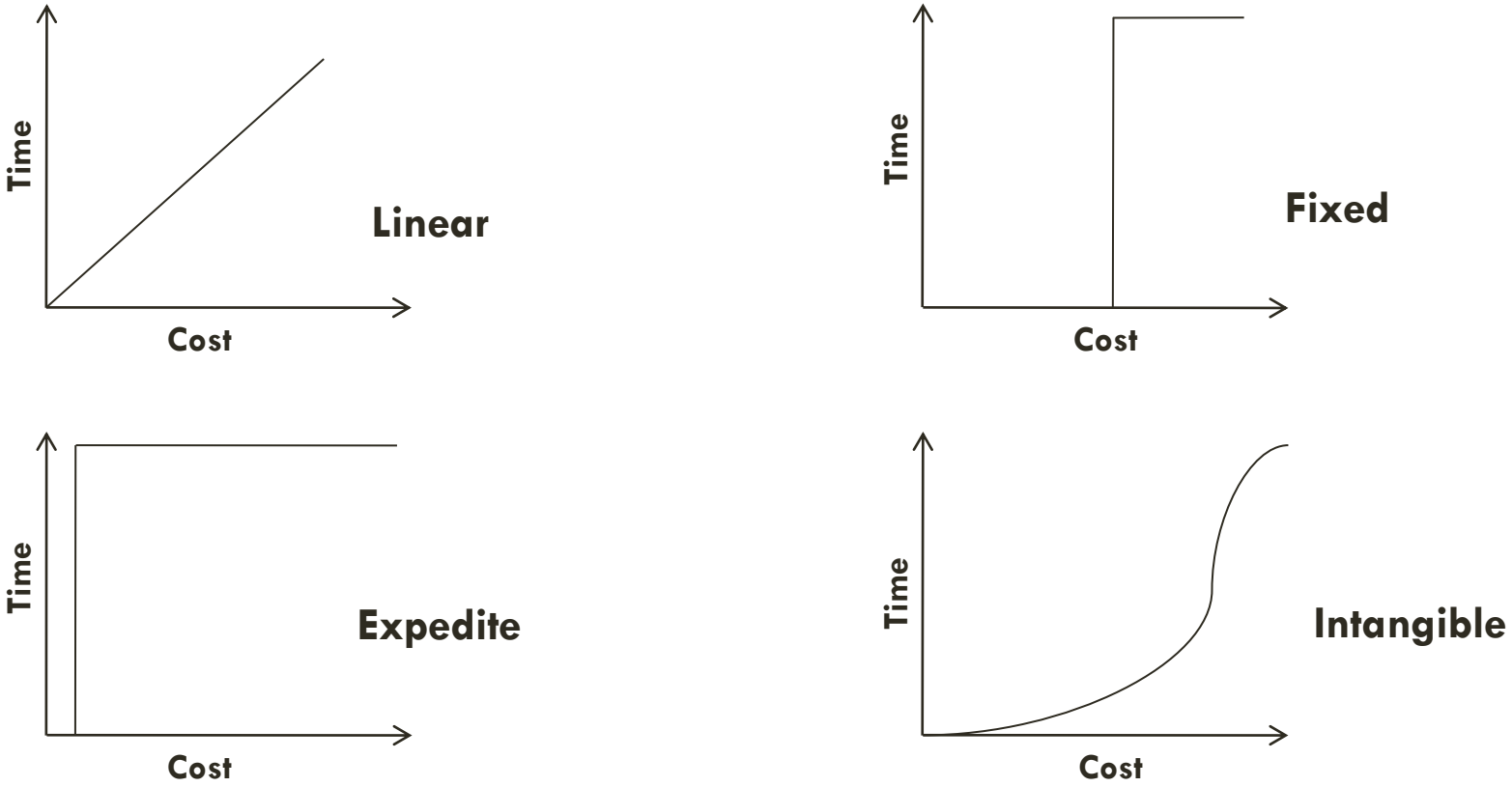
- ***Identify your dev process***
  - How are features decided?
  - What are the various steps involved in materializing it?
- ***Define start and end points for the board***
  - Identify your boundaries
  - Identify when a task enters the board
  - Identify the end of its life cycle on the board

LET'S GET STARTED (CONTD.)

Agree

- Initial WIP limits and policies – can change later
- Prioritization and selection policies
- Policies for different classes of service (expedite, fixed delivery date, standard, intangible)
- Process review cycle time

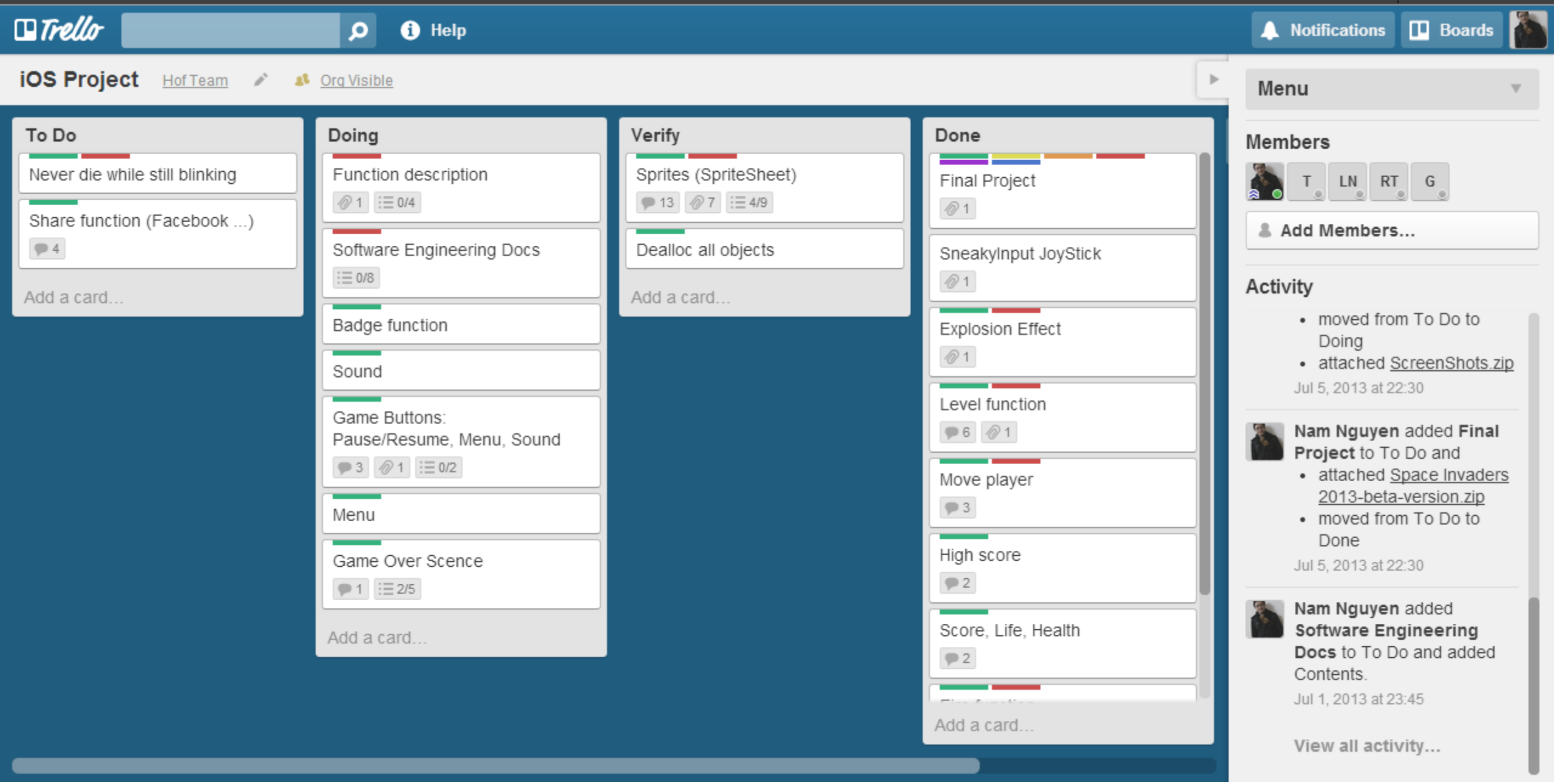
. . . BUT BEFORE GOING ON . . .



Classes of service vs. Cost of Delay

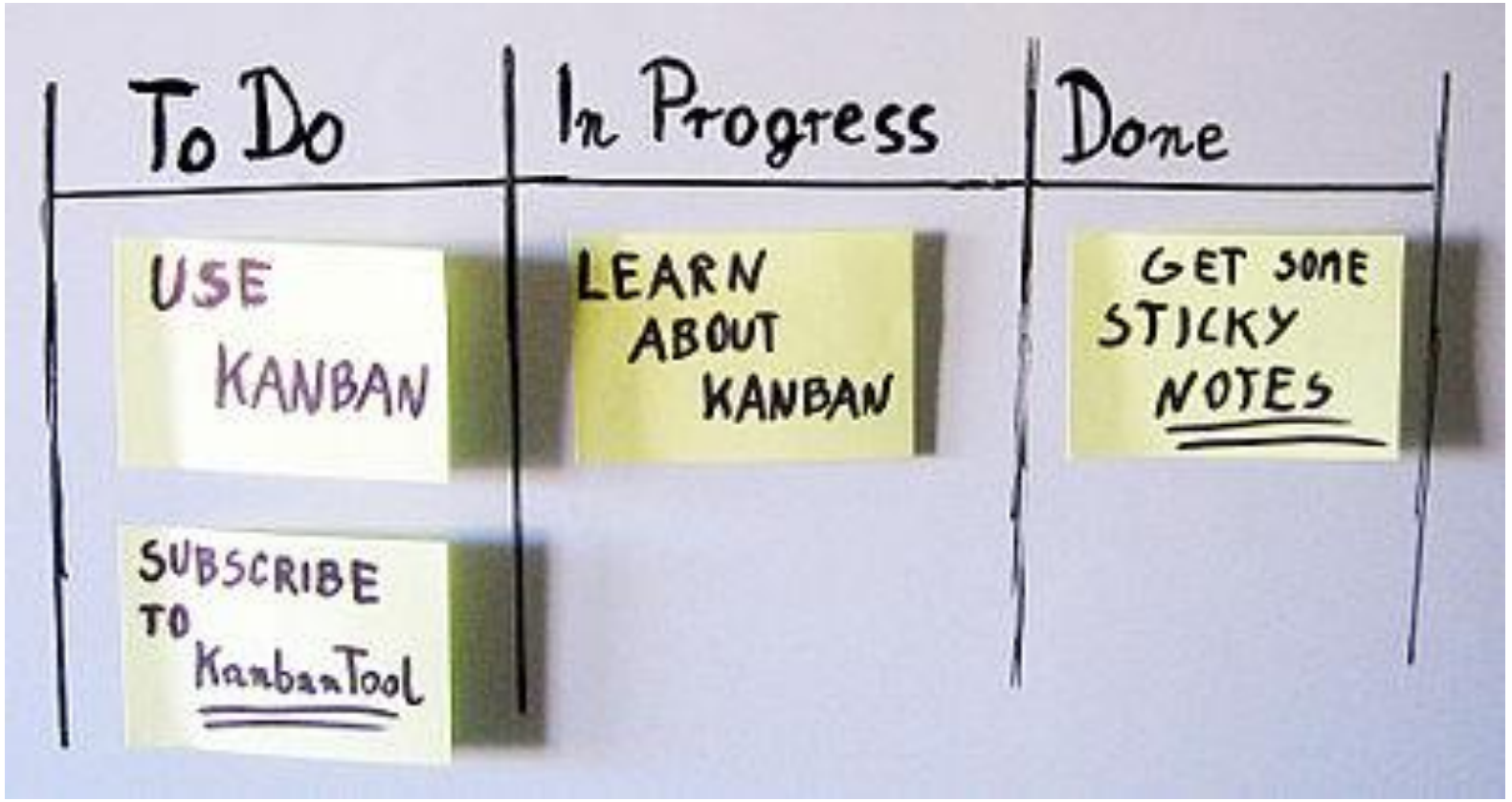
SOME TOOLS TO APPLY KANBAN

- Board and notes: Of course
- https://trello.com
- https://kanbantool.com
- https://kanbanflow.com



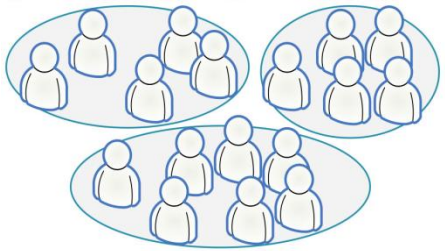
Trello: Kanban + Scrum: tasks for every 1-2 weeks

LET'S GET STARTED

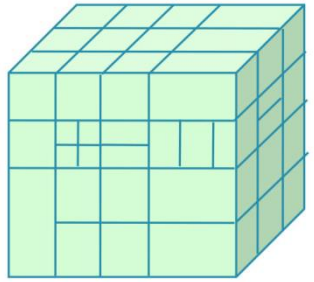


Scrum in a nutshell

Split your organization



Split your product

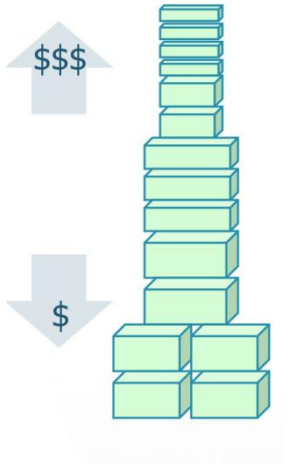


~~Large team~~ spending ~~a long time~~ building a ~~huge thing~~

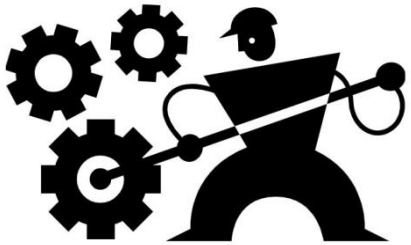
Small team spending a little time building a small thing

… but integrating regularly to see the whole

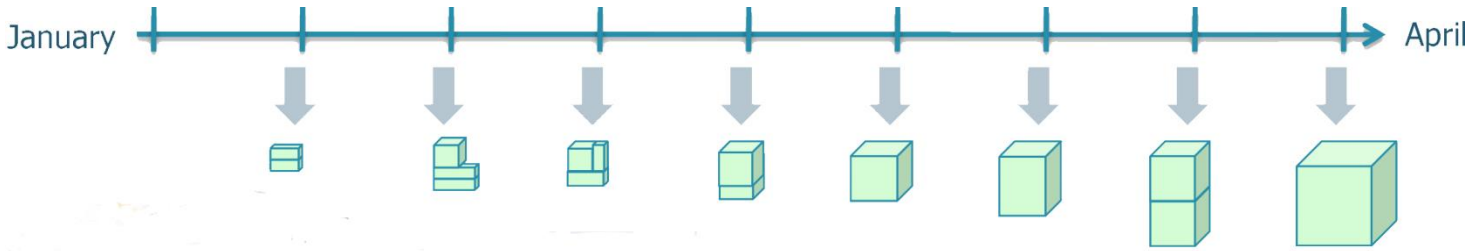
Order the backlog



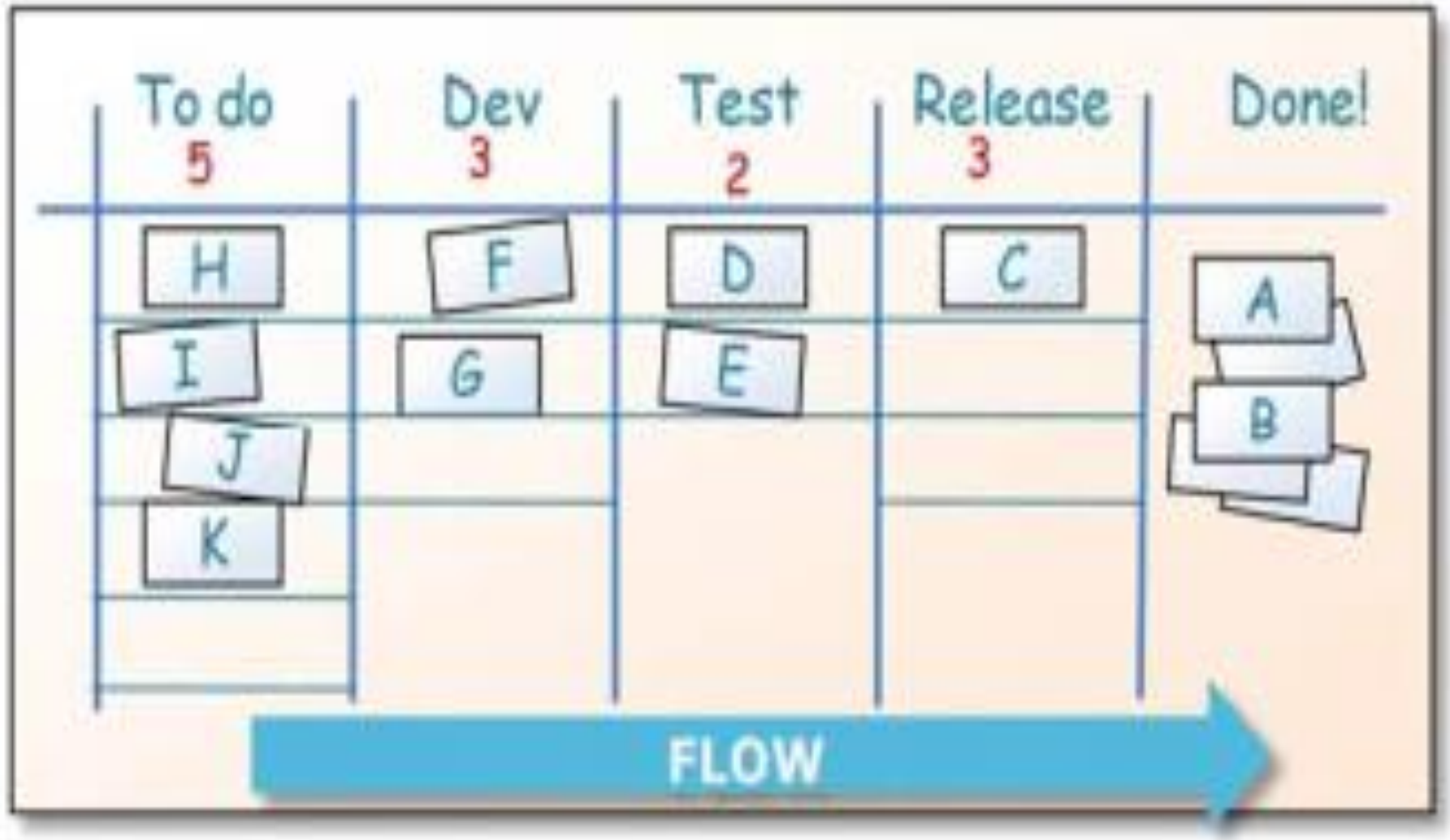
Optimize your process



Split time

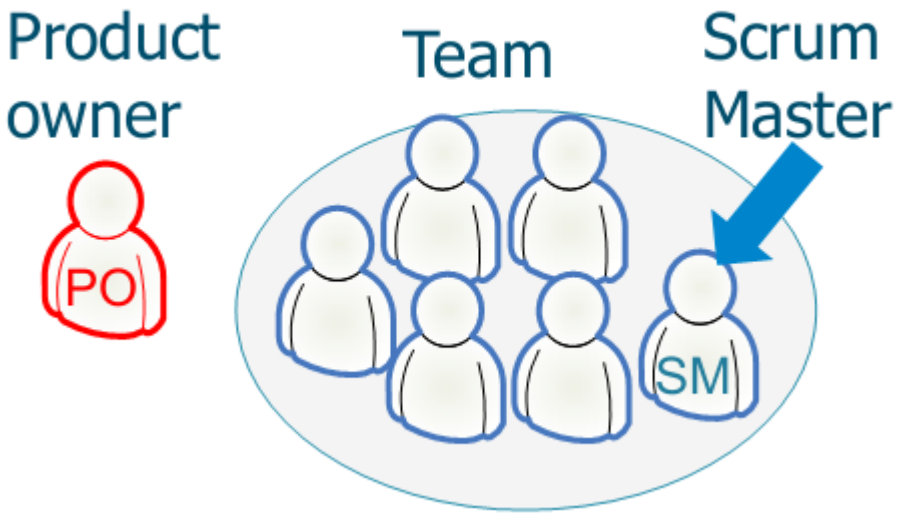


Kanban in a nutshell

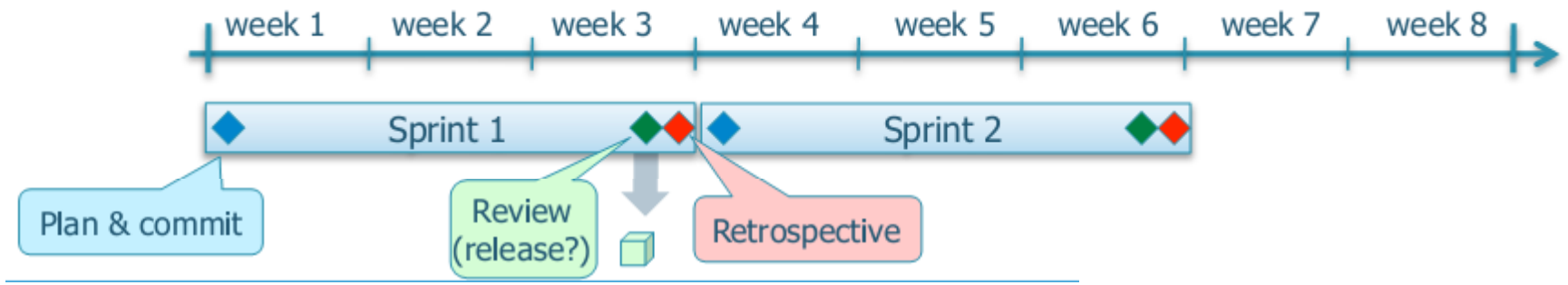

To do
5
Dev
3
Test
2
Release
3
Done!
H
I
J
K
F
G
D
E
C
A
B
FLOW

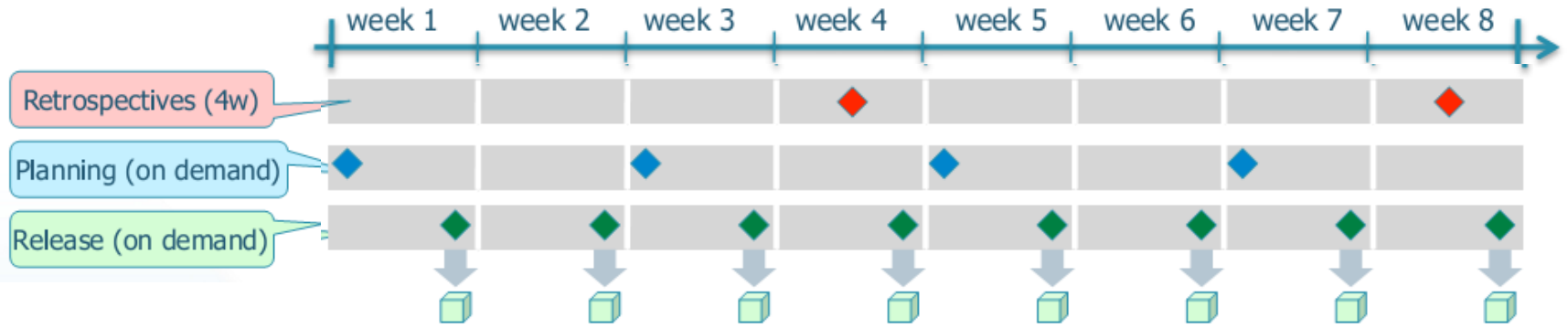
Scrum vs. Kanban

SCRUM PRESCRIBES ROLES, KANBAN DOESN'T!



SCRUM PRESCRIBES TIME-BOXED ITERATIONS

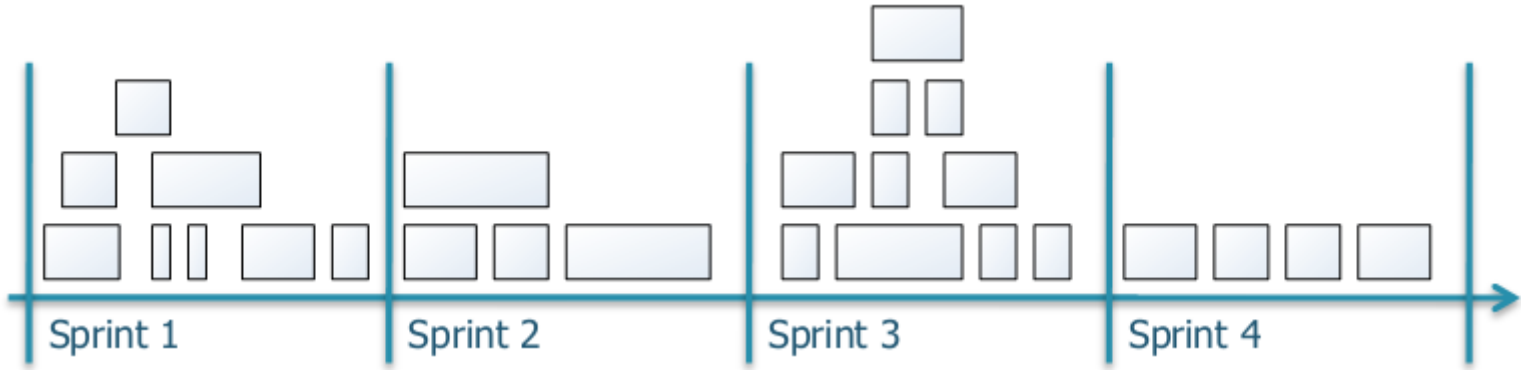




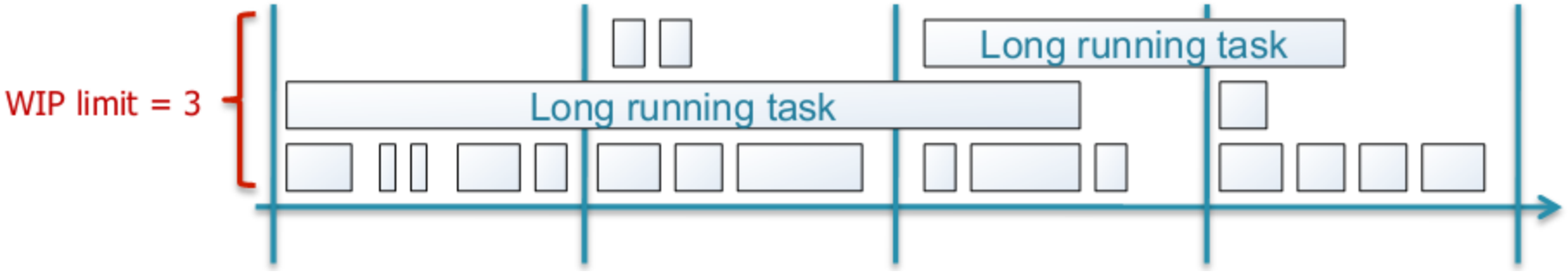
Kanban Team

SCRUM BACKLOG ITEMS MUST FIT IN A SPRINT

Scrum

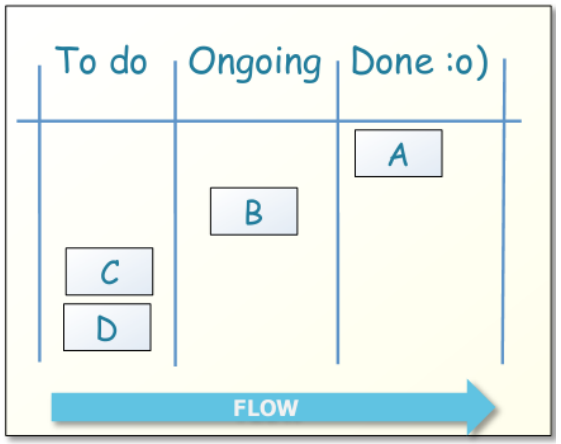


Kanban



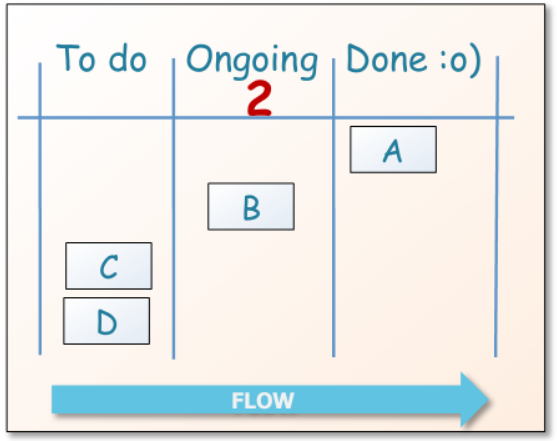
BOTH LIMIT WIP IN DIFFERENT WAYS

Scrum Board



WIP limited per unit of time (iteration)

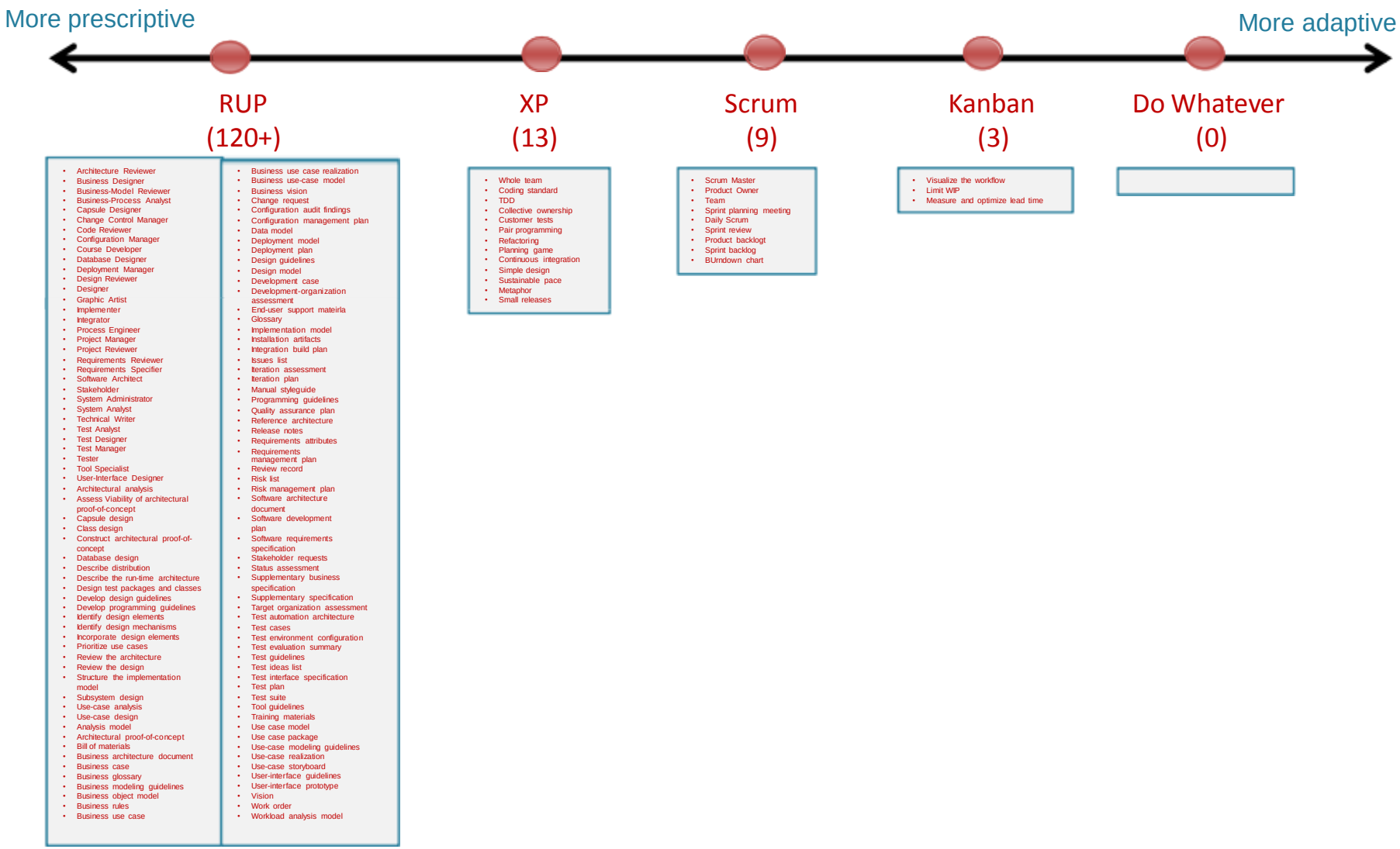
Kanban Board



WIP limited per workflow state

DOES IT MATTER?

- Emphasis should be on the goal and not the tool. Becoming/agile lean is not the goal
- Don't be dogmatic about your process
- There is no good or bad tool. Only good or bad decisions.
- Keep experimenting for understanding and not judgment
- Process is not important, improving the process is important


More prescriptive
More adaptive
RUP
(120+)
XP
(13)
Scrum
(9)
Kanban
(3)
Do Whatever
(0)
Architecture Reviewer
Business Designer
Business-Model Reviewer
Business-Process Analyst
Capsule Designer
Change Control Manager
Code Reviewer
Configuration Manager
Course Developer
Database Designer
Deployment Manager
Design Reviewer
Designer
Graphic Artist
Implementer
Integrator
Process Engineer
Project Manager
Project Reviewer
Requirements Reviewer
Requirements Specifier
Software Architect
Stakeholder
System Administrator
System Analyst
Technical Writer
Test Analyst
Test Designer
Test Manager
Tester
Tool Specialist
User-Interface Designer
Architectural analysis
Assess Viability of architectural proof-of-concept
Capsule design
Class design
Construct architectural proof-of-concept
Database design
Describe distribution
Describe the run-time architecture
Design test packages and classes
Develop design guidelines
Develop programming guidelines
Identify design elements
Identify design mechanisms
Incorporate design elements
Prioritize use cases
Review the architecture
Review the design
Structure the implementation model
Subsystem design
Use-case analysis
Use-case design
Analysis model
Architectural proof-of-concept
Bill of materials
Business architecture document
Business case
Business glossary
Business modeling guidelines
Business object model
Business rules
Business use case
Business use case realization
Business use-case model
Business vision
Change request
Configuration audit findings
Configuration management plan
Data model
Deployment model
Deployment plan
Design guidelines
Design model
Development case
Development-organization assessment
End-user support mateirla
Glossary
Implementation model
Installation artifacts
Integration build plan
Issues list
Iteration assessment
Iteration plan
Manual styleguide
Programming guidelines
Quality assurance plan
Reference architecture
Release notes
Requirements attributes
Requirements management plan
Review record
Risk list
Risk management plan
Software architecture document
Software development plan
Software requirements specification
Stakeholder requests
Status assessment
Supplementary business specification
Supplementary specification
Target organization assessment
Test automation architecture
Test cases
Test environment configuration
Test evaluation summary
Test guidelines
Test ideas list
Test interface specification
Test plan
Test suite
Tool guidelines
Training materials
Use case model
Use case package
Use-case modeling guidelines
Use-case realization
Use-case storyboard
User-interface guidelines
User-interface prototype
Vision
Work order
Workload analysis model
Whole team
Coding standard
TDD
Collective ownership
Customer tests
Pair programming
Refactoring
Planning game
Continuous integration
Simple design
Sustainable pace
Metaphor
Small releases
Scrum Master
Product Owner
Team
Sprint planning meeting
Daily Scrum
Sprint review
Product backlogt
Sprint backlog
BUrndown chart
Visualize the workflow
Limit WIP
Measure and optimize lead time

REFERENCES

- David J Anderson, *Kanban - Successful Evolutionary Change for your Technology Business,* 1st ed, Blue Hole Press, 2010
- Henrik Kniberg, 2009, "*Kanban and Scrum – Making the Most of Both*", Online, Available: http://goo.gl/oiqPG
- Images from *www.kanbantool.com/kanban-analytics-and-metrics*