TECHNISCHE UNIVERSITÄT
CHEMNITZ

Current Trends in Web Engineering

Prof. Dr.-Ing. Martin Gaedke
Dr.-Ing. Sheeba Samuel

Technische Universität Chemnitz

Fakultät für Informatik

Verteilte und selbstorganisierende Rechnersysteme

Chapter://2

KANBAN

Kanban

*a method for defining, managing, and improving **services** that deliver **knowledge work**, such as professional services, creative endeavors, and the design of both physical and software products*

(Andersen and Carmichael 2016)

- Inspired by Toyota Production System (TPS) and Lean principles
- Start with what you are doing now, then improve
- Development as pipelines with service level agreements

Kanban Concepts

Service

*One or more people collaborating to produce (usually intangible) work products for a **customer** who requests the work and who accepts or acknowledges delivery of the completed work.*

(Andersen and Carmichael 2016)

Flow System

A system characterized by the entry and departure of work items. It is a way of viewing knowledge work by the flow of items from the request or idea through to its delivered value.

(Andersen and Carmichael 2016)

Lean

Improvement through elimination of waste

Section://2

Kanban Overview

Kanban Roles

There are no required roles, but 2 roles emerged from practice:

Service Request Manager

- responsible for understanding the needs and expectations of customers
- responsible for facilitating selecting and ordering work items at the Replenishment Meeting
- (somewhat) comparable to Product Owner

Service Delivery Manager

- responsible for the flow of work in delivering selected items to customers
- responsible for facilitating the Kanban Meeting and Delivery Planning
- (somewhat) comparable to Scrum Master

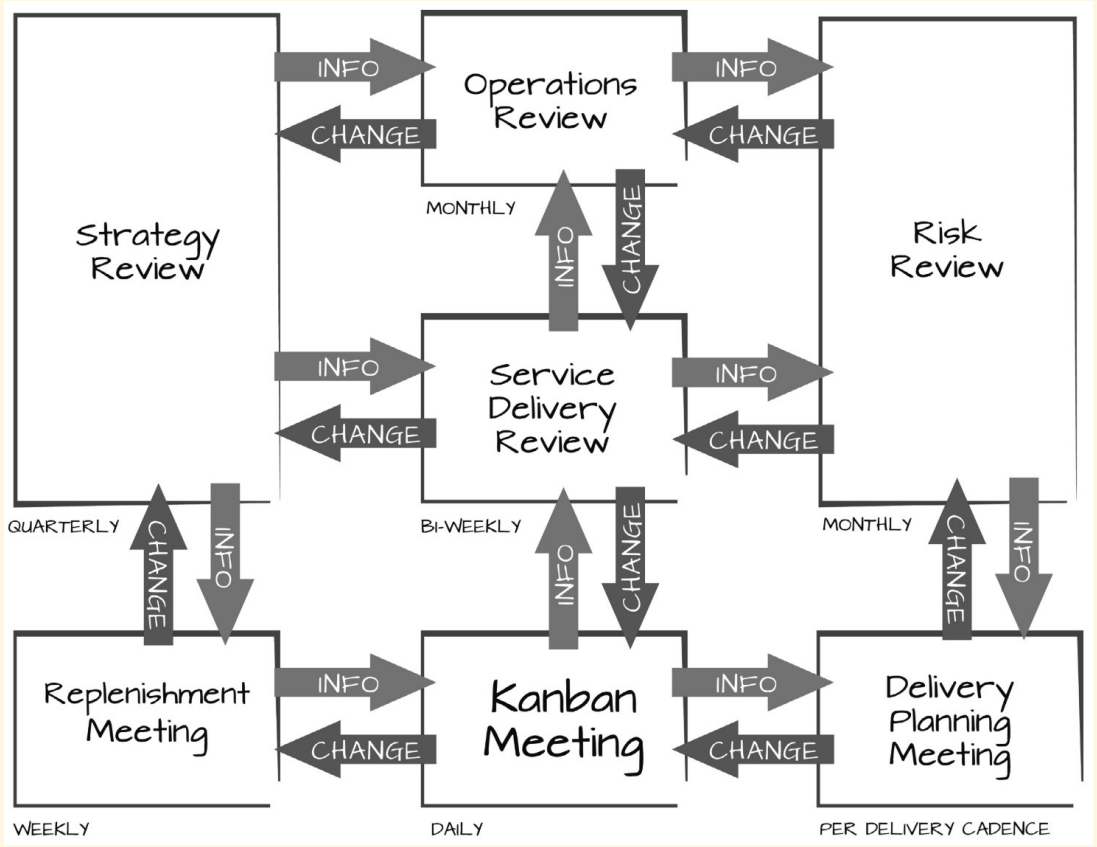
Kanban Core practices (1)

1. Visualize work
   - Kanban board with process stages and work items
2. Limit work in progress
   - pull from previous stage
   - WIP limits per stage
3. Make policies explicit
   - Transition rules between stages (who, when, what)

Kanban Core practices (2)

4. Manage flow
   - measure length of queues, times per stage, throughput
   - provide planning information
5. Implement feedback loops
   - Retrospectives, standups
6. Improve collaboratively, evolve experimentally
   - process improvements using theory of constraints, system thinking, PDCA etc.

Cadences



(Andersen and Carmichael 2016)

Kanban Baseline Meetings

Kanban Meeting

- Daily standup meeting
- daily coordination, self-organization, and planning review
- Focus on completing work items and unblocking issues

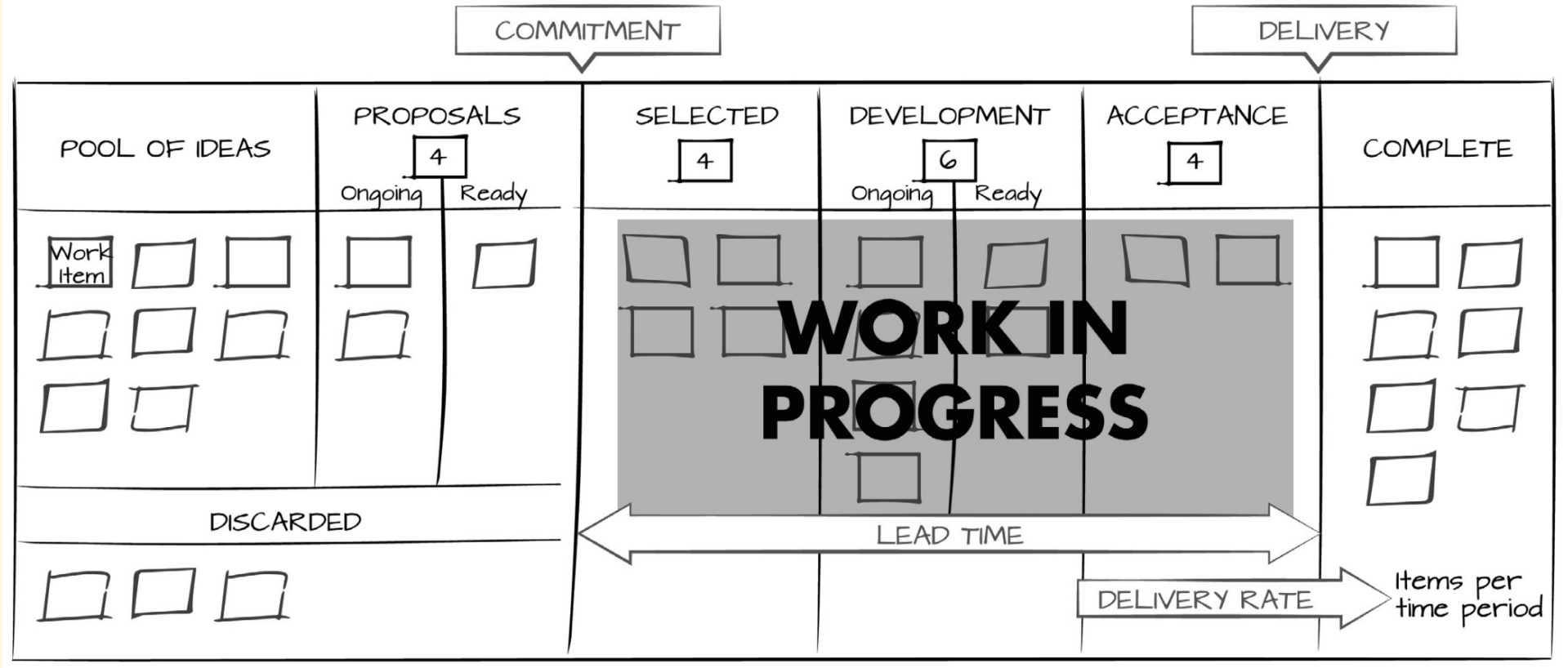
Replenishment Meeting

- replenishing the input queue with work items
- prioritization/commitment: from proposed work items to “should”
- collaborative prioritization

Kanban Board

- Visualization of work
- Columns represent stages, cards represent work items
- Often: Todo, Doing, Done
- Visible work in progress (WIP) limits per column
- Sometimes: Fast-track lanes

Kanban Board



(Andersen and Carmichael 2016)

Kanban Board

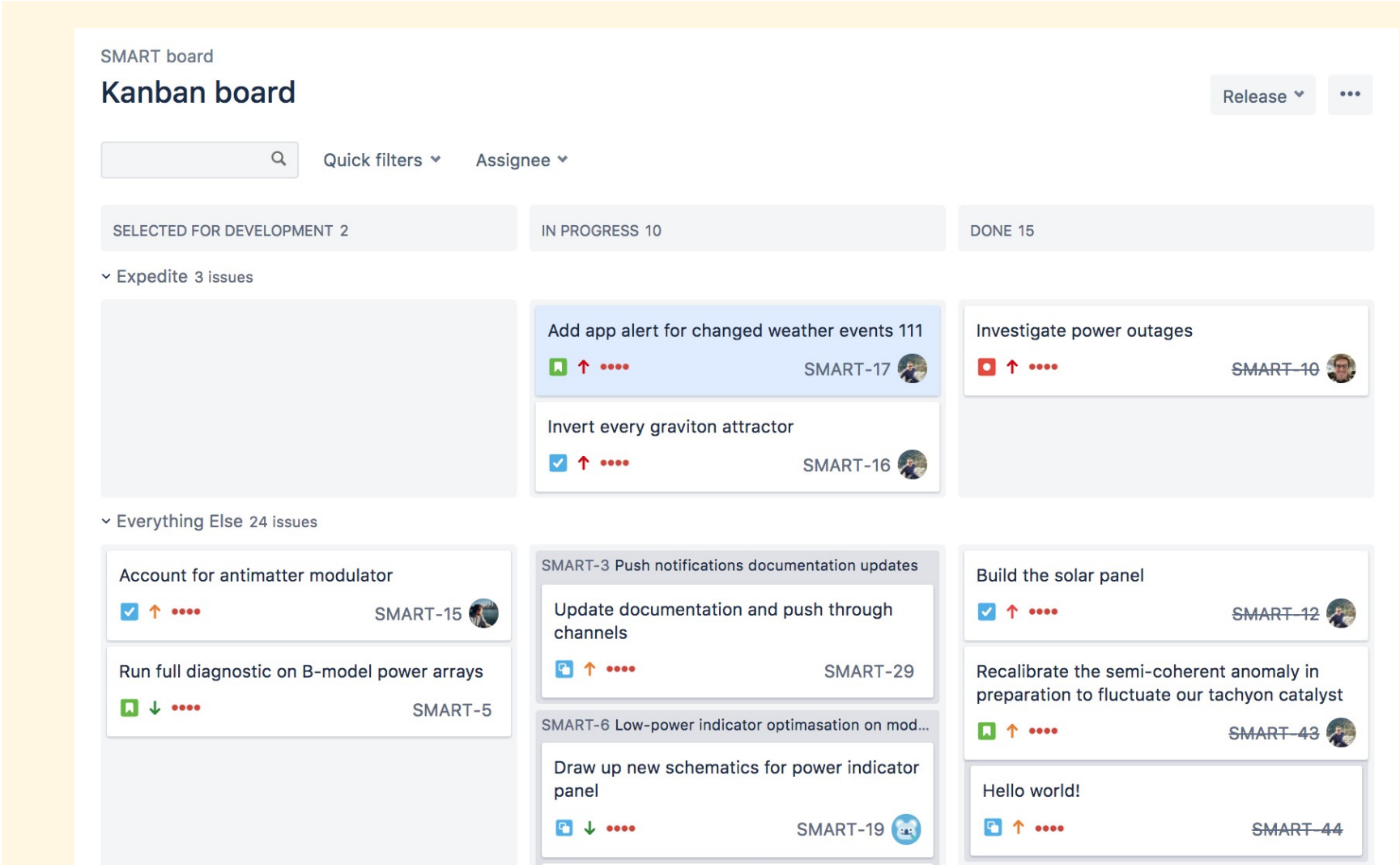


Image Source: https://confluence.atlassian.com/jirasoftwarecloud/monitoring-work-in-a-kanban-project-764478148.html

Kanban vs Scrum

Similarities

- pull systems
- process visibility
- continuous releases of small software increments
- value continuous improvements and adaptions

Differences

- no fixed iterations, continuous delivery instead of sprint releases
- focus on continuous flow, less planning
- optional role model, not necessarily cross-functional teams
- Cycle/lead time instead of velocity as base measure for improvement
- stronger focus on lean, improvements

Kanban Applicability

In comparison to other agile software development approaches, it is particularly applicable to

- Software that requires continuous delivery
- Companies having problems with effort estimation and interruptions (support, maintenance, bug fixing)
- Maintenance and Bugfixes in dedicated teams
- IT Operations
- Program/Portfolio level
- Outside software: marketing, recruitment, organizational strategy

Discussion

- Why is it important to visualize work and progress in Kanban with regard to flow?
- How long should an iteration in Kanban be?
- Why is there less planning in Kanban compared to Scrum and what are the consequences?
- Provide arguments why the focus on improvement is stronger in Kanban compared to Scrum.
- What would be the decision criteria if you had to decide to use Kanban or Scrum?