

manage software

agile philosophy. kanban framework

Panin Alexei

recap - previous lecture

**what is the role most close to people and
what is the role the most close to business ?**

Scrum Master
Project Delivery Manager
Tech Project Manager
Project Manager
Proxy Product Owner
Business Analyst
Product Owner
Product Manager

people
business

what is management triangle ?



what is the reality of project management ?

make the plan
execute the plan
expect the plan to change
adapt to the new plan

**reality of
project
management**

list most common project limitations

goal
target audience
budget
time

project
limitations

list most common project dependencies

partners
third parties
legal third parties
administrative third parties

**project
dependencies**

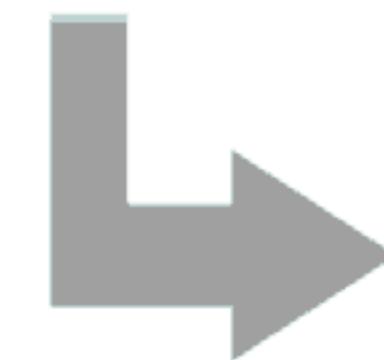
list most common project risks

budget changes
human capital
legal changes
project scope changes

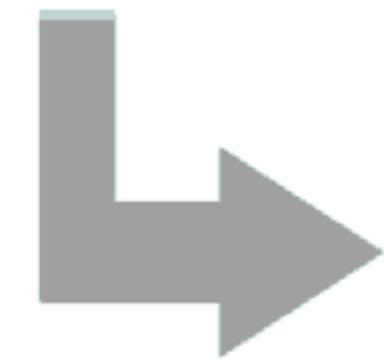
**risk
management**

list phases of waterfall

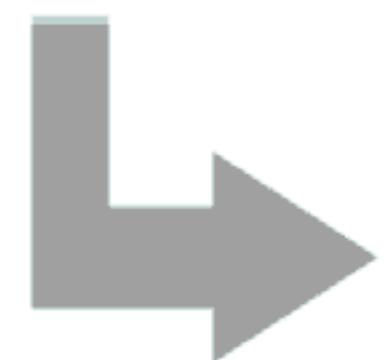
Requirements



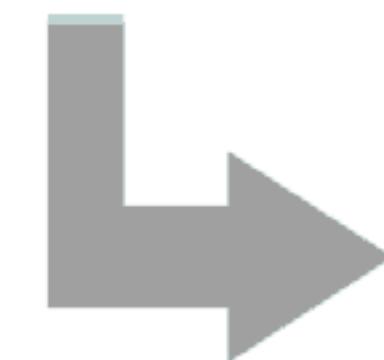
Design



Execution



Verification



Deployment

list project phases

short description
project goal
requirements and expectations
limitations: time, budget, target
audience, other resources
dependencies

**project
brief**

**essential - never skip
documented
conclusions**

research

project brief
roadmap
implementation plan
execution plan
budget
team
success metrics

technical plan

sign all documents
visibility on the project
plan for all stakeholders
adjust roadmap

project
kickoff

regular meetings
visibility on progress / reporting
risk assessment
stakeholder management
plan adjustments and revision of goal

**project
execution**

project sign off
project closure report
feedback

project
closure

the formal birth of agile

In February 2001, a group of seventeen software developers, including luminaries like Kent Beck, Martin Fowler, and Ward Cunningham, met at a ski resort in Utah, USA.

In February 2001, a group of seventeen software developers, including luminaries like Kent Beck, Martin Fowler, and Ward Cunningham, met at a ski resort in Utah, USA.

They shared their experiences and frustrations with traditional software development methodologies and sought to define a new approach that prioritizes flexibility, collaboration, and customer satisfaction.

In February 2001, a group of seventeen software developers, including luminaries like Kent Beck, Martin Fowler, and Ward Cunningham, met at a ski resort in Utah, USA.

They shared their experiences and frustrations with traditional software development methodologies and sought to define a new approach that prioritizes flexibility, collaboration, and customer satisfaction.

The result of this meeting was the Agile Manifesto, a brief statement of values and principles for Agile software development. It emphasized individuals and interactions, working software, customer collaboration, and responding to change over following a plan and rigid processes.

agile manifesto

Individuals and interactions over
processes and tools

agile
manifesto

**Individuals and interactions over
processes and tools**
**Working software over
comprehensive documentation**

**agile
manifesto**

**Individuals and interactions over
processes and tools**

**Working software over
comprehensive documentation**

**Customer collaboration over
contract negotiation**



**agile
manifesto**

Individuals and interactions over processes and tools

Working software over comprehensive documentation

Customer collaboration over contract negotiation

Responding to change over following a plan



agile manifesto

12 principles behind the agile manifesto

<https://agilemanifesto.org/principles.html>

**our highest priority is to satisfy the customer
through early and continuous delivery
of valuable software.**

1

welcome changing requirements, even late in development. agile processes harness change for the customer's competitive advantage.

2

deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.

3

business people and developers must work together daily throughout the project.

4

**build projects around motivated individuals.
Give them the environment and support they need,
and trust them to get the job done.**



the most efficient and effective method of conveying information to and within a development team is face-to-face conversation.

6

**working software is the
primary measure of progress.**

7

**agile processes promote sustainable development.
the sponsors, developers, and users should be able
to maintain a constant pace indefinitely.**

8

continuous attention to technical
excellence and good design enhances
agility.

9

**simplicity--the art of maximizing the
amount of work not done--is essential.** 10

**the best architectures, requirements,
and designs emerge from self-
organizing teams.**

11

at regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.

12

management framework

key elements

rules / guidelines / principles

key
elements

rules / guidelines / principles
artifacts / tools / outputs

key
elements

rules / guidelines / principles
artifacts / tools / outputs
roles / responsibilities

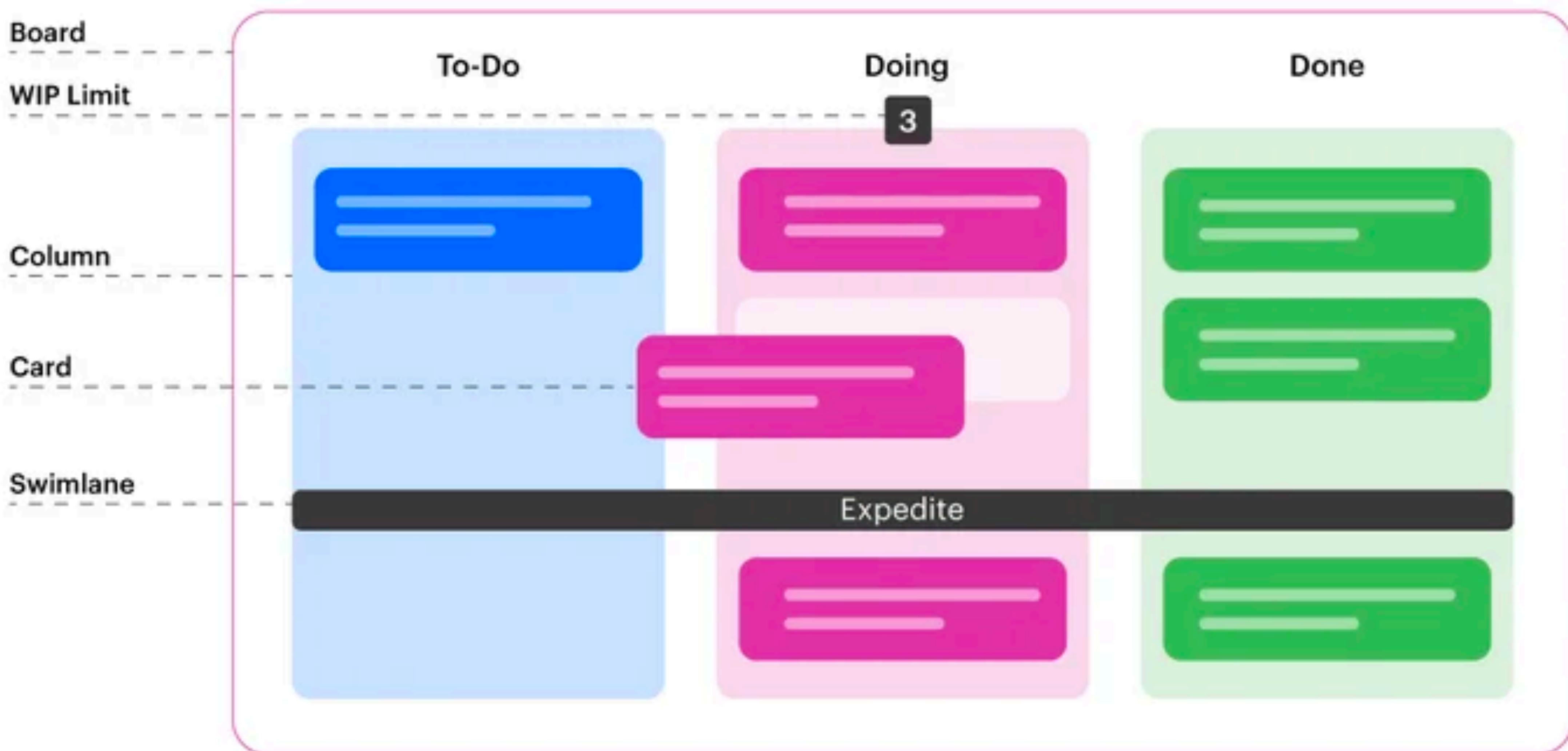
key
elements

rules / guidelines / principles
artifacts / tools / outputs
roles / responsibilities
metrics

key
elements

kanban framework

Kanban board



kanban framework rules

visualize the workflow

rules

visualize the workflow
limit Work in Progress (WIP)

rules

visualize the workflow
limit Work in Progress (WIP)
manage flow

rules

visualize the workflow
limit Work in Progress (WIP)
manage flow
make process policies explicit

rules

visualize the workflow
limit Work in Progress (WIP)
manage flow
make process policies explicit
improve collaboratively

rules

kanban framework artefacts

Kanban Board

Flight reservation platform

Backlog 1/2	Backlog 2/2	Selected	To specify		In development		To test	To deploy in production	
			In progress	Done	In progress	Done	Min 1	In progress	Done
ProjectId_54 Activities search bar Research	ProjectId_50 Yield management Price&Payment	Flights	ProjectId_49 Destination to anywhere Offer	Min 1 Done	ProjectId_37 Show public holidays in local countries Research	ProjectId_36 Price alert feature Price&Payment	Min 1 In progress	ProjectId_23 Adapt cookies to GDPR Account	Max 3 Done
			ProjectId_39 Facebook SSO Account		ProjectId_38 Adding several travelers to a unique account Account	ProjectId_28 Multi-city flights Research	Max 2 Done	ProjectId_24 Paypal integration Price&Payment	
			ProjectId_48 Airline filter Research		ProjectId_40 Historical flight activity (5 years back) Account	ProjectId_29 Show number of tickets left Offer		ProjectId_25 Seat selection Offer	
ProjectId_55 Customer support through Messenger Account	ProjectId_51 SMS confirmation 24h before departure Account	Other services	ProjectId_43 Car rental special offers Offer		ProjectId_42 Car rental Offer	ProjectId_41 Shuttle and transfer information Research		ProjectId_27 Proposing accommodation at arrival place Offer	
									ProjectId_21 Hotel search bar Research
ProjectId_56 Android App Account	ProjectId_52 Flight research time optimization Research	Bugs	ProjectId_45 Connection airports display Research		ProjectId_44 Currency conversion Price&Payment	ProjectId_33 Check-in baggage information Account	ProjectId_30 Multi-page flight display Offer	ProjectId_31 Integrity of Air France data Offer	
			ProjectId_46 Autocompletion with special characters Research				ProjectId_32 Number of tickets left error Offer		
ProjectId_57 iOS App Account	ProjectId_53 Arrival time feature Research	Fridge	ProjectId_47 Privacy policy update Account		ProjectId_34 Vaccine passport upload Account	ProjectId_35 Last minute offers Offer			



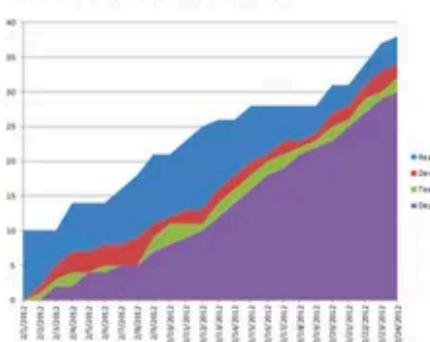
To discuss at next meeting

Data integrity of last minute flight offers

Call on a GDPR expert

Determine the list of currencies taken into account by the system

Cumulative flow chart



kanban board

1

The Kanban board is the primary artifact used to visualize the workflow. It typically consists of columns representing different stages of work (e.g., To Do, In Progress, Done) and cards representing individual work items. The board provides a visual representation of work in various states and helps team members understand the flow of work through the system.

work items (cards)

2

Work items, often represented as cards on the Kanban board, are the individual tasks, user stories, or features being worked on by the team. Each card contains relevant information about the work item, such as its title, description, assignee, priority, and status.

WIP limits

3

Work In Progress (WIP) limits are constraints placed on the number of work items allowed in each column of the Kanban board. WIP limits help prevent overburdening the team or the system, reduce multitasking, and optimize flow efficiency.

backlog

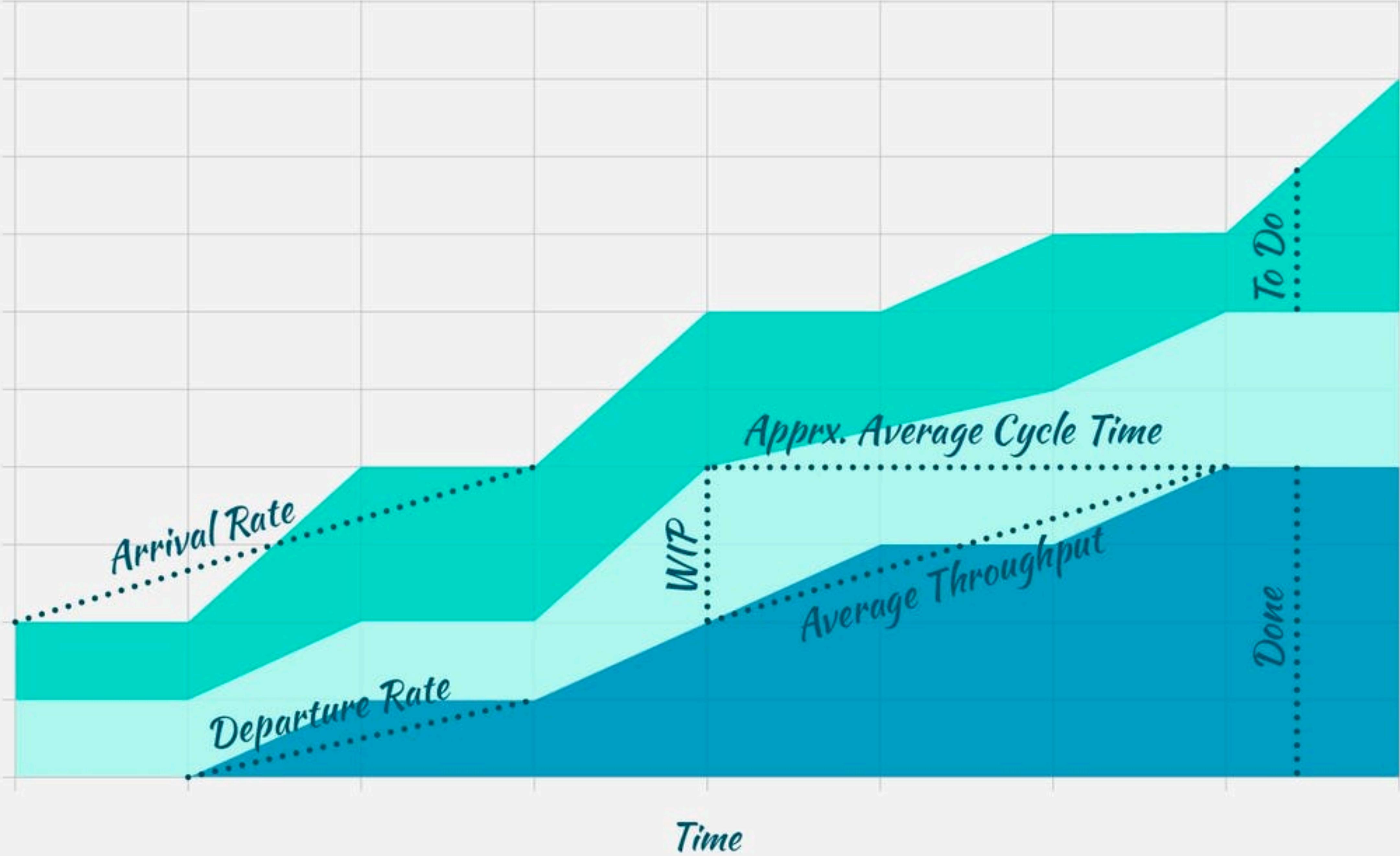
4

The backlog is a list of all pending work items that have not yet been started. It serves as a repository for new ideas, feature requests, bugs, or tasks awaiting prioritization and scheduling. The backlog helps teams maintain a clear view of upcoming work and prioritize tasks effectively.

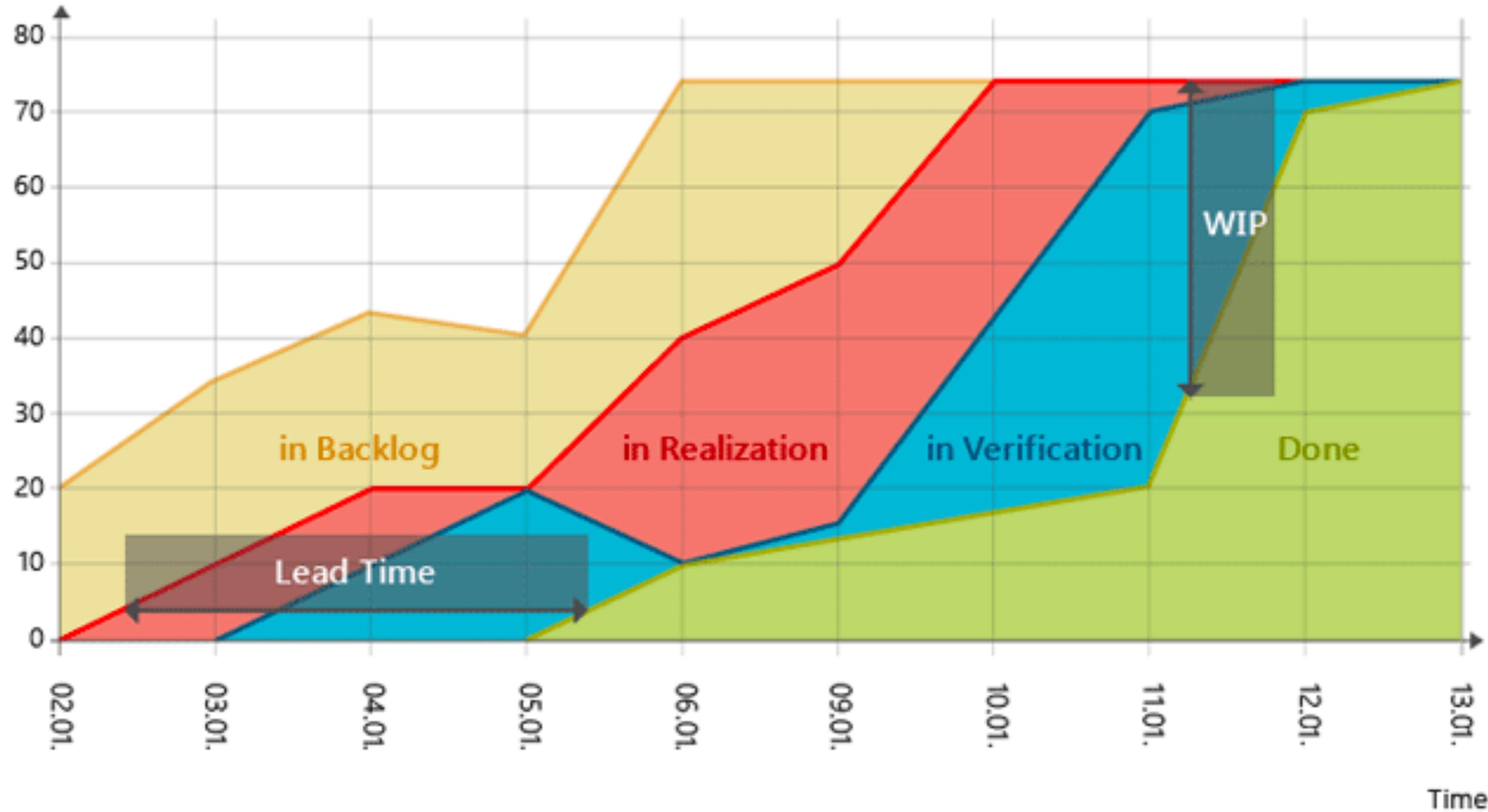
cumulative flow diagram (CFD) 5

The Cumulative Flow Diagram is a graphical representation that shows the flow of work items through the Kanban system over time. It tracks the number of work items in each stage of the workflow at various points in time, providing insights into the stability and efficiency of the process.

Tasks



Requirements
(Number / Effort)

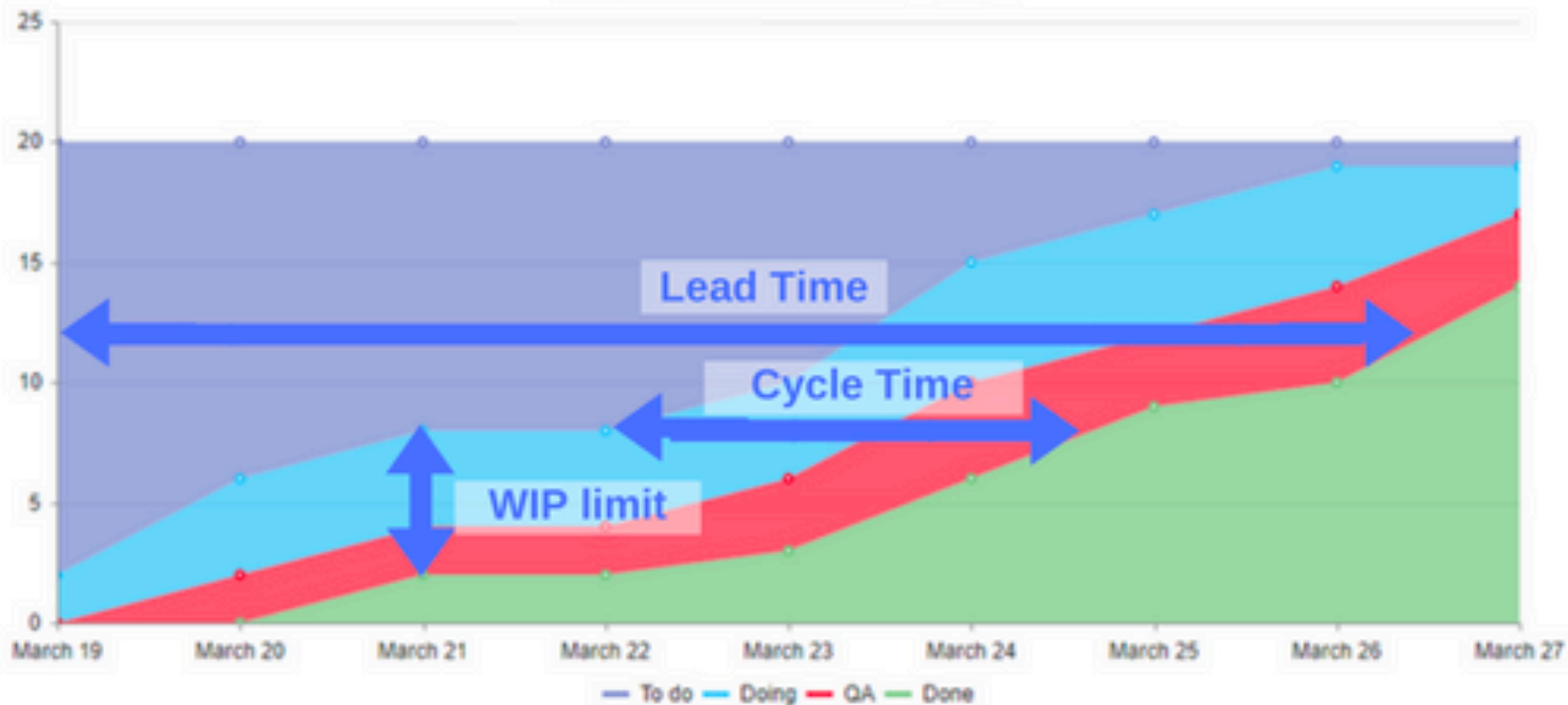


Cumulative Flow chart type

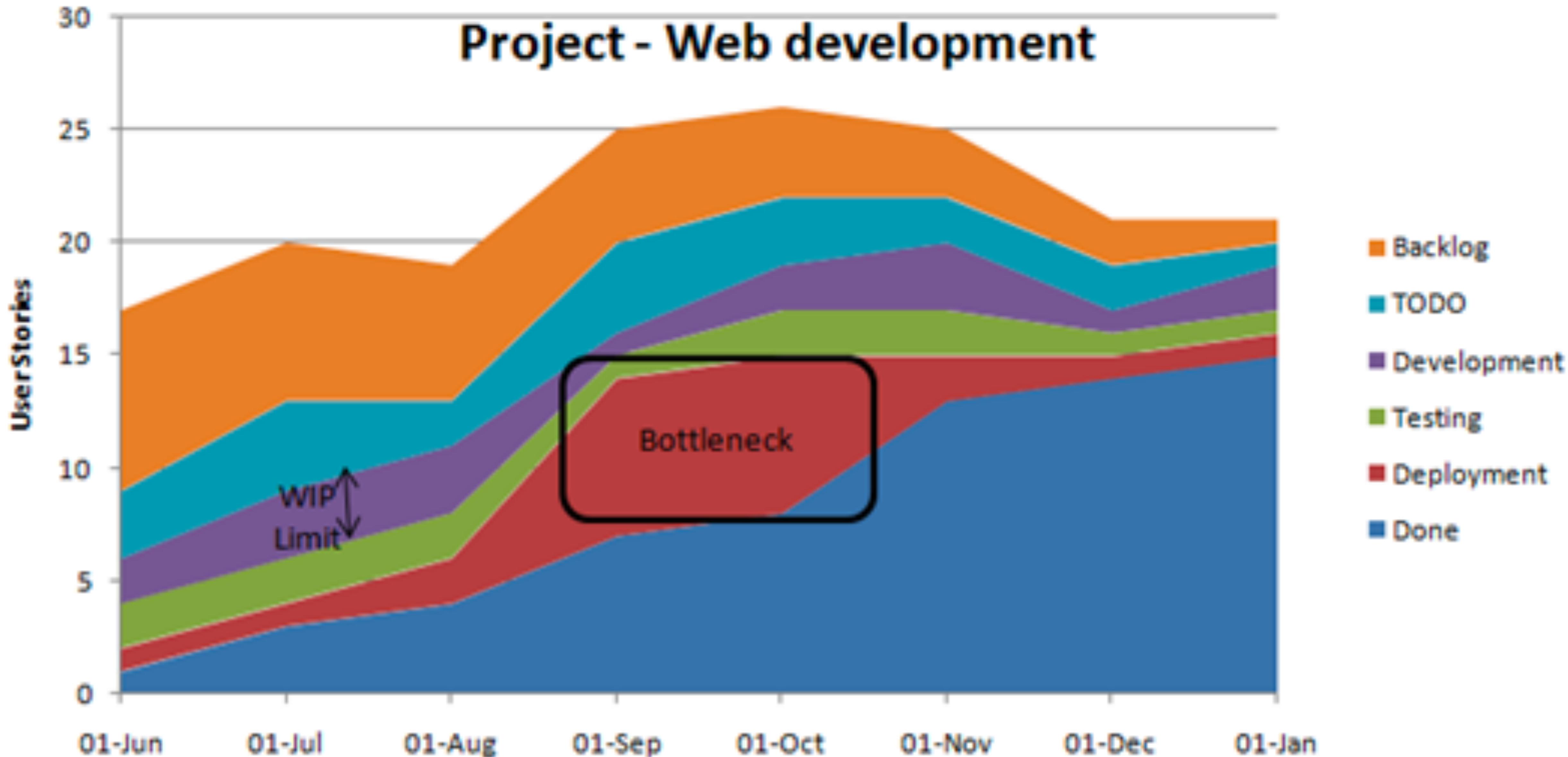
Story

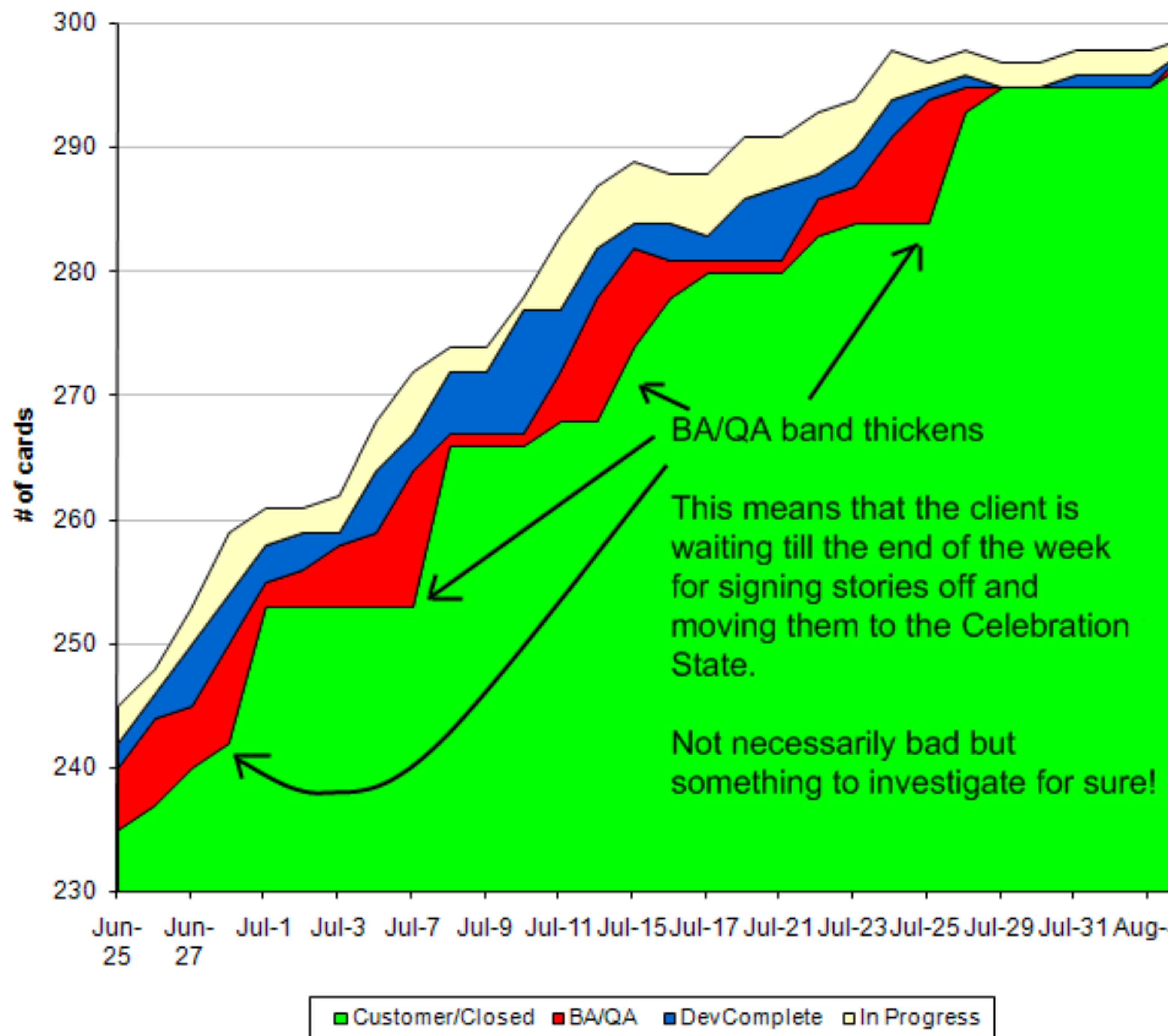
Refresh

Cumulative Flow Chart by story



Project - Web development

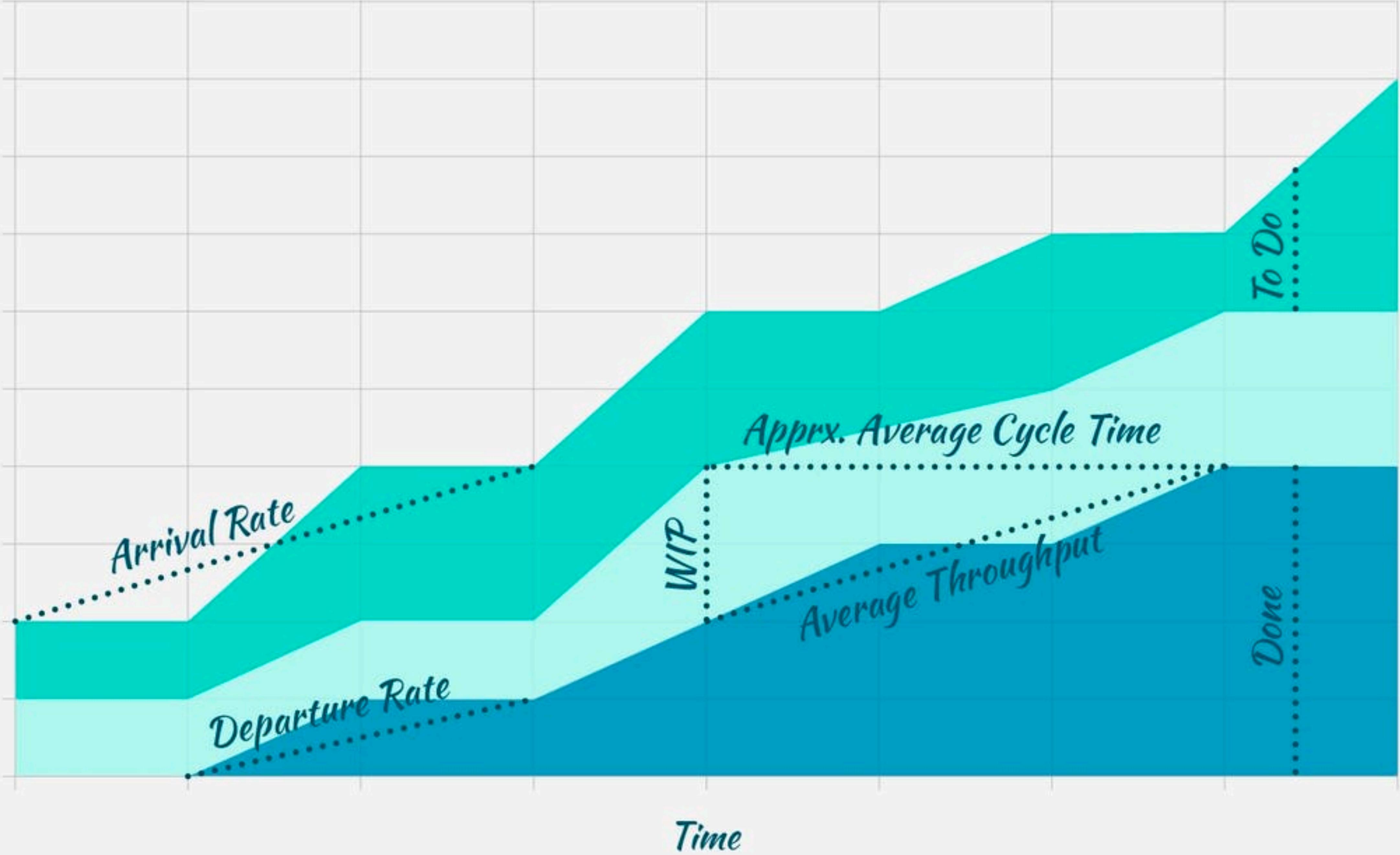




metrics and performance data 6

Various metrics and performance data, such as cycle time, lead time, throughput, and WIP trends, serve as artifacts in Kanban. These metrics help teams monitor performance, identify bottlenecks, and make data-driven decisions about process improvements.

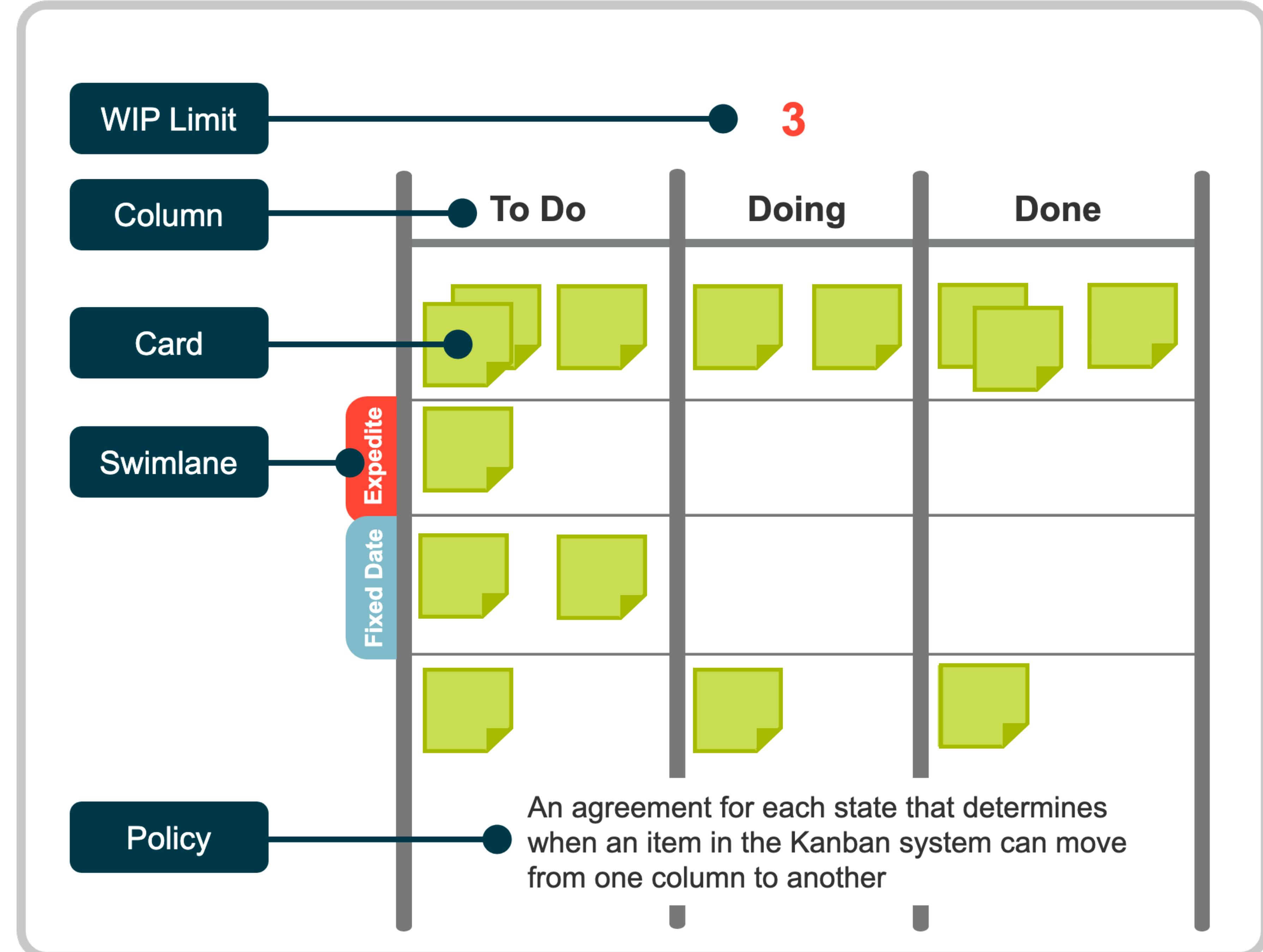
Tasks



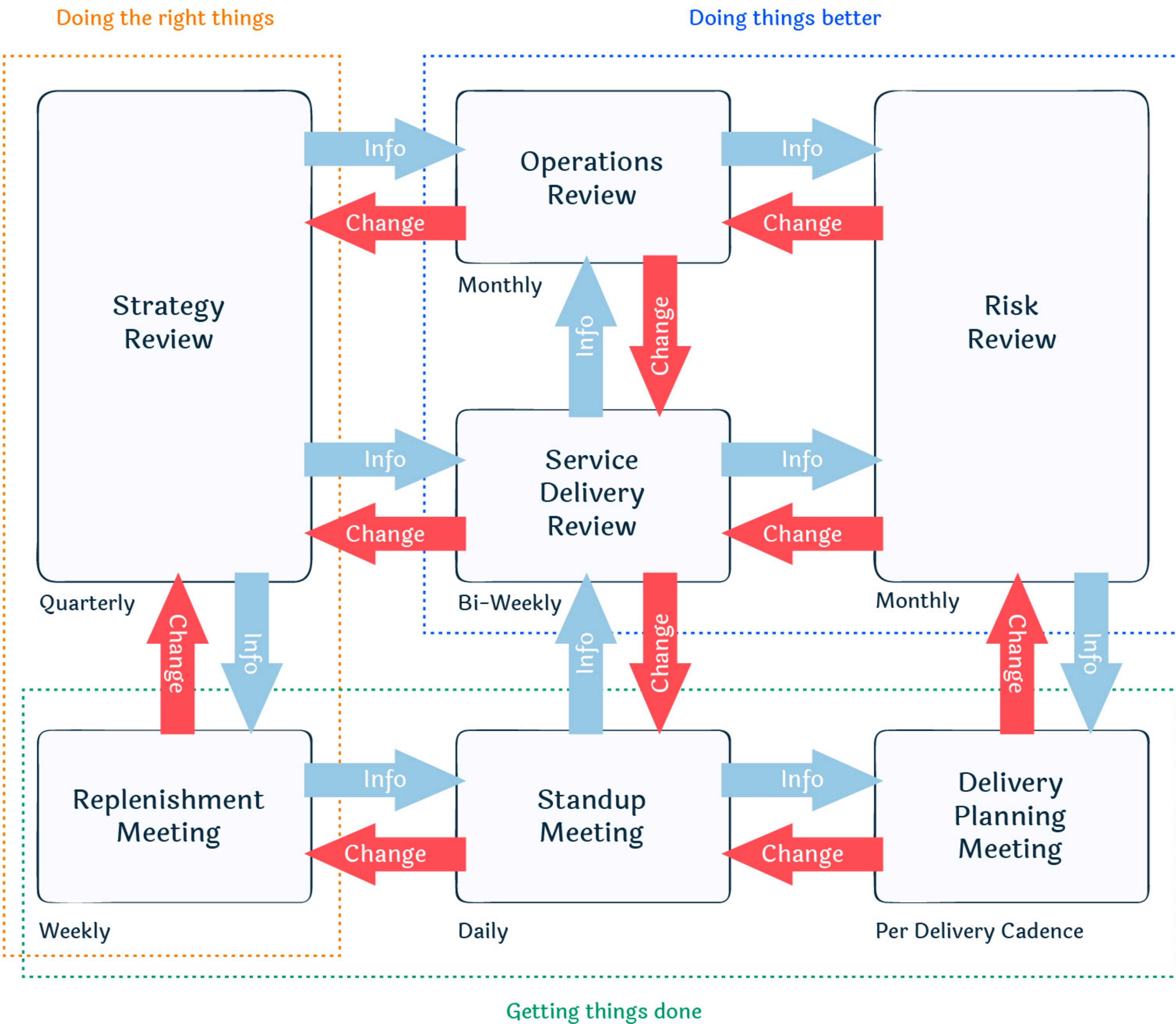
policies and work item classes

7

Policies and work item classes define rules, guidelines, or criteria for how work items are prioritized, assigned, processed, and completed within the Kanban system. They help ensure consistency, clarity, and alignment of practices across the team.



kanban framework events



kanban framework use-cases

<https://resources.scrumalliance.org/Article/scrum-vs-kanban>

practical assignment

CFD



workflow simulation