ACIT Hub

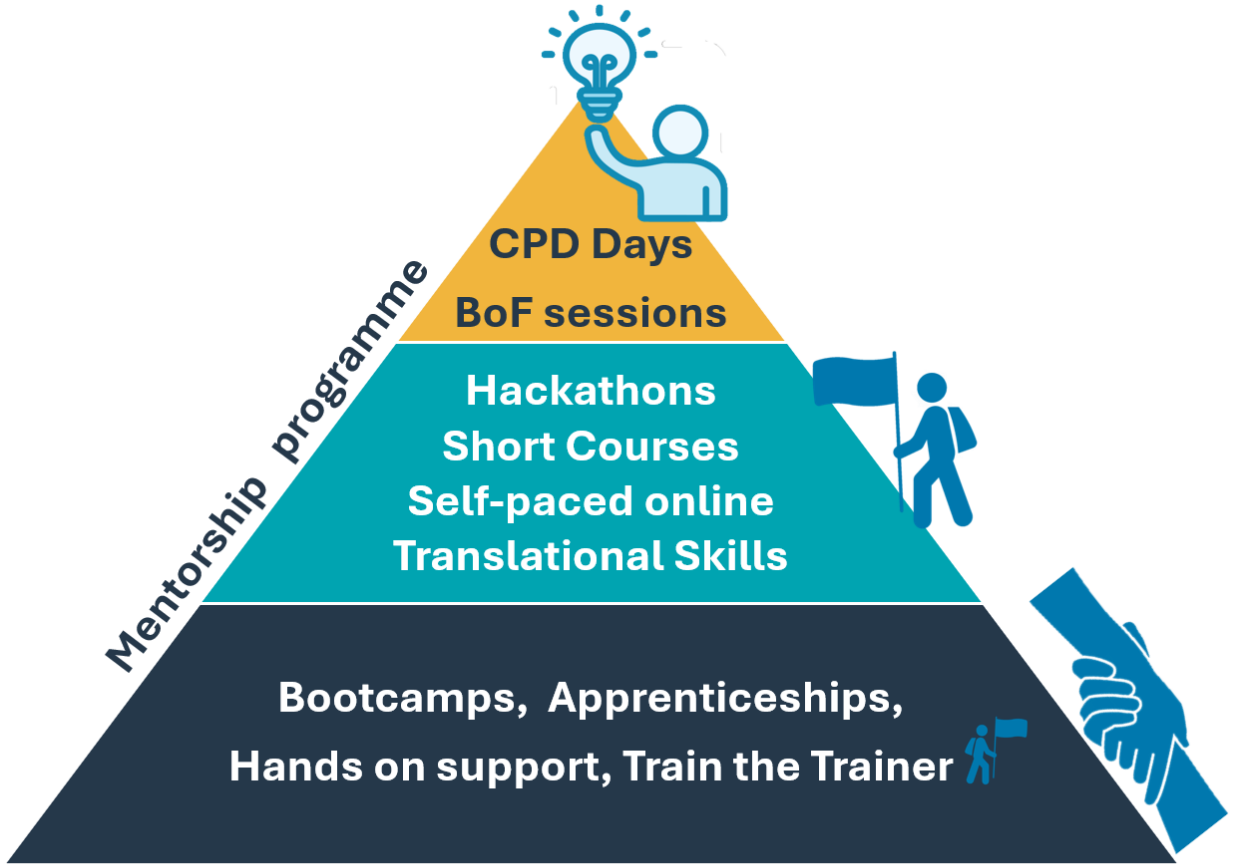
WELCOME TO THE ACIT HUB WEBINAR SERIES

UK Research and Innovation

Engineering and Physical Sciences Research Council

What is ACIT Hub?

- UKRI DRI Training Hub for RTP's
  - Research Technology Professionals
  - £3.9m total
- Aims – Focus on RIE's
  - Train & Share best practice
  - Raise profile
  - Improve career paths
  - Foster engagement with DRI funded initiatives (Tier 0/1/2 HPC)



Upcoming training & events

HPC Days 2026 Durham 15th-19th June

- RIE's & RSE working together towards user friendly, sustainable HPC
- Demystifying the Scheduler

Next webinar: July

Next hackathon: September

Agile in Research Computing Workshop

- RIE-focused Agile training course later in October
- In collab with a DisCouRSE-funded project

Co-design days

Housekeeping

- All microphones will be muted during the webinar
- Please ask questions via the Q&A box (not the chat box)
- Use the  if there's a question you would like to see answered
- Code of conduct - https://acit.surrey.ac.uk/events/codeofconduct.html
- EDI survey
- Feedback for today's webinar

Today's speaker

Dr. Jaydeep Mody, Team Lead - Research Computing Sys Admin and ACIT-Hub Co-Lead, IT Services, University of Surrey

With a PhD in Computational Physics, Jaydeep started as an RTP in 2013. He is currently a Team Lead of the Sys Admin team within Research Computing Services at the University of Surrey. His activities include a mix of line/project management, coaching/ mentoring, and providing software and systems support for Linux-based HPC and Research IT Platforms.

Becoming Agile in Research Computing – A Learning Journey

Dr Jaydeep Mody

Team Lead – Research Computing Sys Admin

University of Surrey, IT Services

(20th May 2026)

Welcome and Important Information

- Session will aim to focus on "lessons learnt" in Agile Journey

- Feel free to engage
  - Audience poll questions with options
  - Ask questions using Q & A section
  - Don't be shy with emojis - 👍 😂 ❤️

- No certified Agile professionals were involved – content is based purely on an ongoing "Learning Journey" through trial/error, plenty of reading/research, and shared experience/feedback from colleagues

Audience poll...

Typical Research-IT Services

The problems?

- Unaware of much Agile or need for any structured framework for workflow management - work processed in an ad-hoc manner – not all the time, but on many occasions prioritised based on who shouts loudest, emergencies, artificial priorities, mood, interests, etc.

- Work arriving through tickets, individual inboxes, meetings, corridor chats, etc. – leading to “unpleasant” phrases heard more frequently than normal.

    - ***“Sorry, I could not get “X” done as I have way too much on my plate, and I am feeling quite stressed…”***

    - ***“Oh yes, we discussed (promised) “X” in the meeting we had 3 months ago…remember?! ”***

    - ***“He/She had emailed me a month ago about “X”…sorry…it’s due in 2 days!”***

    - *(Typically by Managers)* ***“Oh! Thought you were working on that already for the past X days/weeks?”***

Lesson 1 – Buy-in and Mindset

- 2 important prerequisites for implementing a functional structured workflow process like Agile
- Not going too far if you do not have buy-in from upper management. “At the very least” – this is needed from your immediate line manager
- Tried some form of “structured working” prior, but for us, this “buy-in” opportunity truly arose and solidified when covid hit in 2020 and team switched to remote-working
- And if you thought “buy-in” was the toughest part…you would be wrong
- Next task was to help team evolve and make a “mindset shift” – a continuous process

Lesson 2 – Centralising work capture

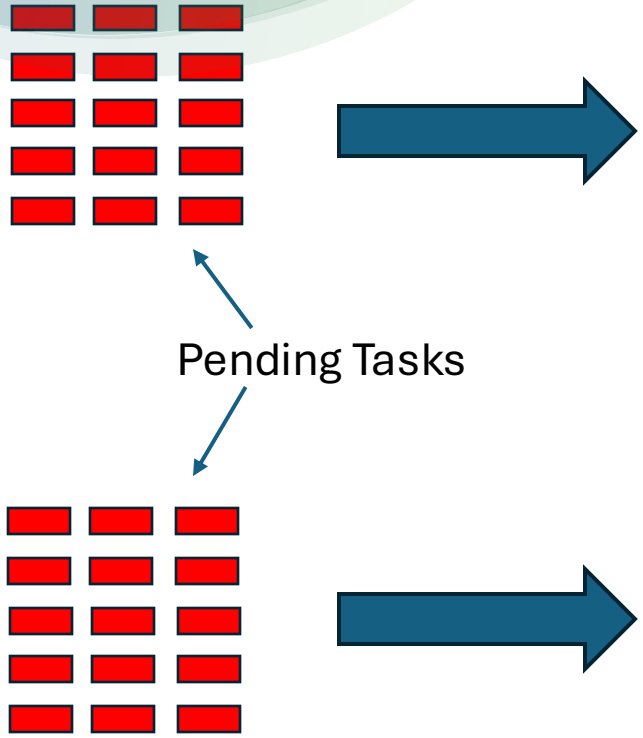
- Agile boards help with this – often called a “Backlog”.
- Started using a “Backlog”, however some work was still coming through multiple channels
- Had to make continuous effort to capture **everything** (however small or big) in **one-place**
    - This central capture cannot be underestimated – serves as the foundation to next steps **(Needs to be backed up by regular review!)**
- Great – we have a way to do this! 2 popular frameworks – Scrum and Kanban.

Lesson 3 – Choosing appropriate Agile approach!

	Scrum	Kanban
Origin	Software Development	Lean Manufacturing
Core Idea	Work is planned and delivered in **fixed-length iterations** called sprints (typically 1–4 weeks), i.e. work is "time-boxed"	Workflow is a **continuous** through the system with no fixed iterations/sprints
Approach	Team **commits to a set** of work items at the start of each sprint. (**Push-based** system)	Work is pulled as capacity becomes available and is **flexible**. (**Pull-based** system)
Roles	Product owner, Scrum master, Development team	No strict roles
Best Suited For	Product development with **structured planning** and regular review cycles	Continuous delivery, support/operational environments with **rapidly changing** priorities
Mindset	Plan a batch of incremental work and complete it within the **sprint**	Limit work in progress and optimise **flow** through the system

Lesson 3 – Choosing appropriate Agile approach!

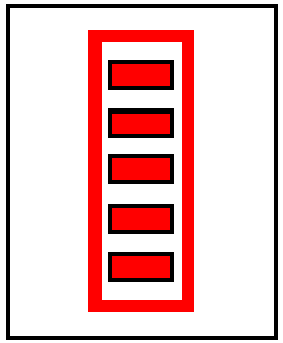
Scrum



Kanban

Lesson 3 – Choosing appropriate Agile approach!

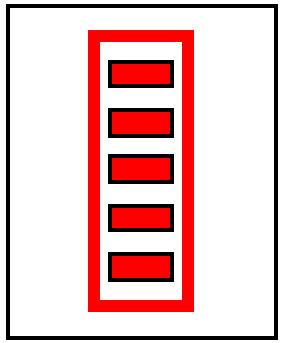
Scrum



Kanban

Lesson 3 – Choosing appropriate Agile approach!

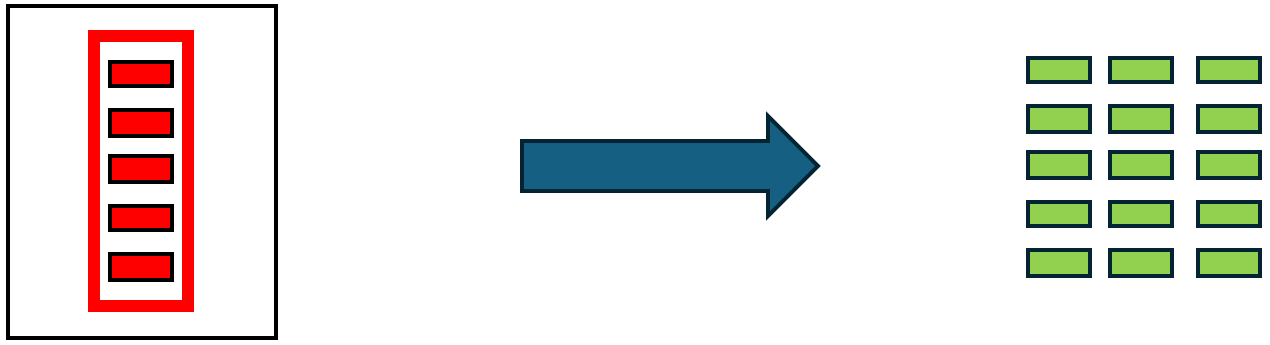
Scrum



Kanban

Lesson 3 – Choosing appropriate Agile approach!

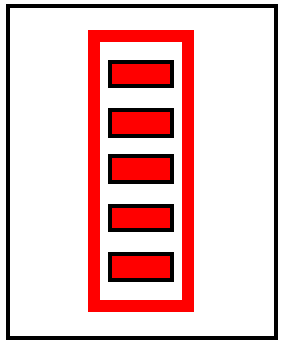
Scrum



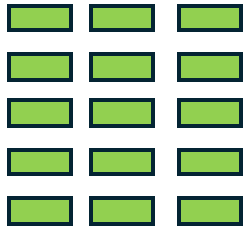
Kanban

Lesson 3 – Choosing appropriate Agile approach!

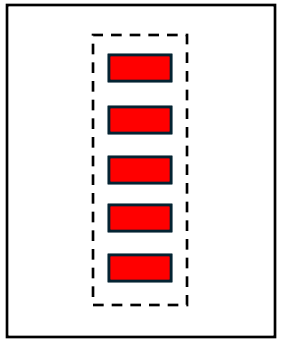
Scrum

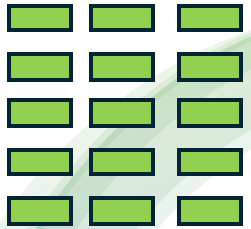


Completed Tasks



Kanban





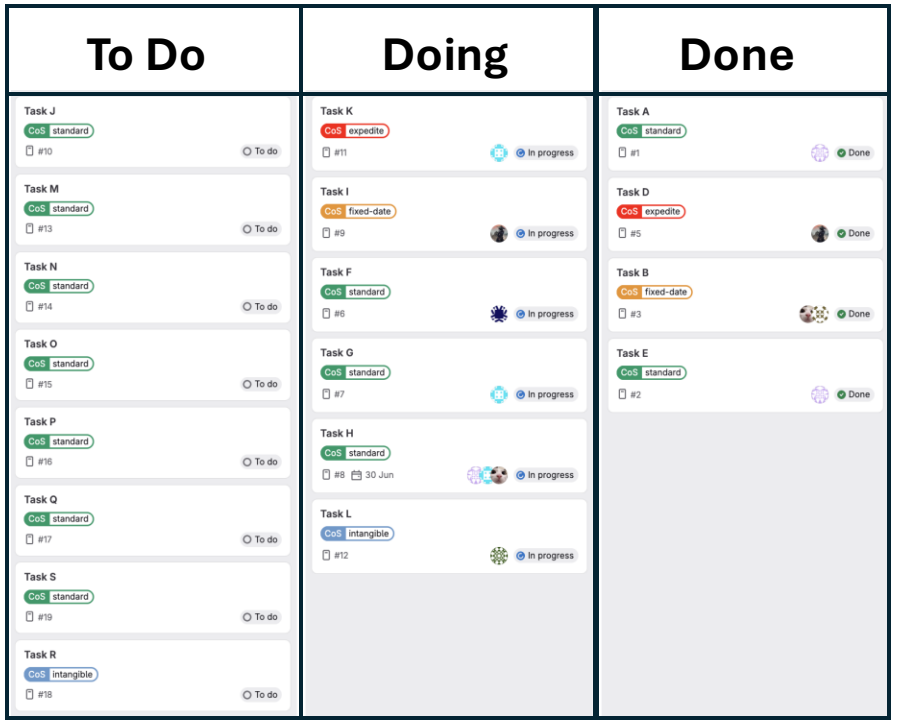
Lesson 3 – Choosing appropriate Agile approach!

- Scrum's potential drawback for RIE (Research Infrastructure Engineering) environments
  - Typically, fixed tasks over a fixed period (1-4 weeks; usually 2 weeks). Avoid other distractions
  - Experimented for very short period (as a quick test) but soon found out its rigidness is not allowing for flexibility, which Kanban clearly offers, so made switch back to Kanban
- Came to conclusion
  - **Scrum** – typically for "development" teams with more planned tasks and structure (eg. RSEs)
  - **Kanban** – typically for "operational" teams with a more dynamic approach and flexibility (eg. RIEs)

Another key Kanban benefit – no "overheads" for initial setup – visualise your current real-time snapshot and iterate as you go!

Lesson 3 – Choosing appropriate Agile approach!

- Kanban – method to visualise workflow, using a “board” with columns
- Columns represent the various stages of work, starting with 3 basic columns, often many more columns.



Lesson 4 – A need for Prioritisation

- Through consistent capturing of “work-items” – overtime, the backlog grew and overwhelmed the boards!

- We had a list of captured work-items, but struggled with which should be picked? Decision paralysis – often picking inappropriately

- To pick based on what?
  - Which user "shouts" loudest?
  - What you "feel like" or "enjoy" the most?
  - Management wants it done?
  - Are there any serious hidden budget/programme deadlines upcoming? How would you know?
  - Oldest task? (Age is good but not always practical – plus how do you know age if not tracked?)
  - Some of this would happen in reality sometimes but "should be the exception, not the rule"

- If not above, can we have some clarity/rules for picking? Felt like we were missing something obvious
  - Researched **prioritisation methods/frameworks**

Lesson 4 – A need for Prioritisation

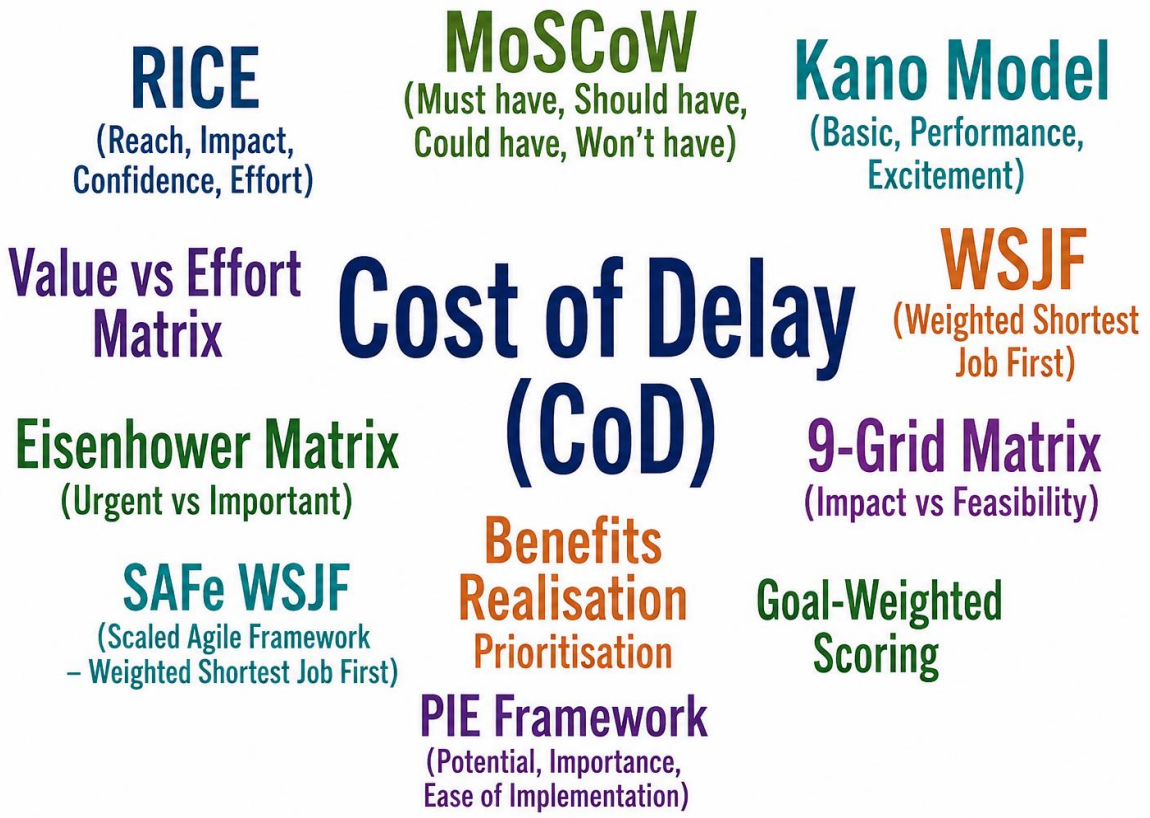
- Kanban **“Classes of Service” (CoS)** – this truly felt like a breakthrough for our situation!
- CoS help categorize different types of work based on nature, priority, delivery times, etc.
- Each CoS represents a specific set of rules/expectations regarding how work items should be treated and processed
- Every work-item assigned to 1 of 4 CoS

1. Expedite
2. Fixed-Date
3. Standard
4. Intangible

But what criteria to use to assign work-items to these classes…?

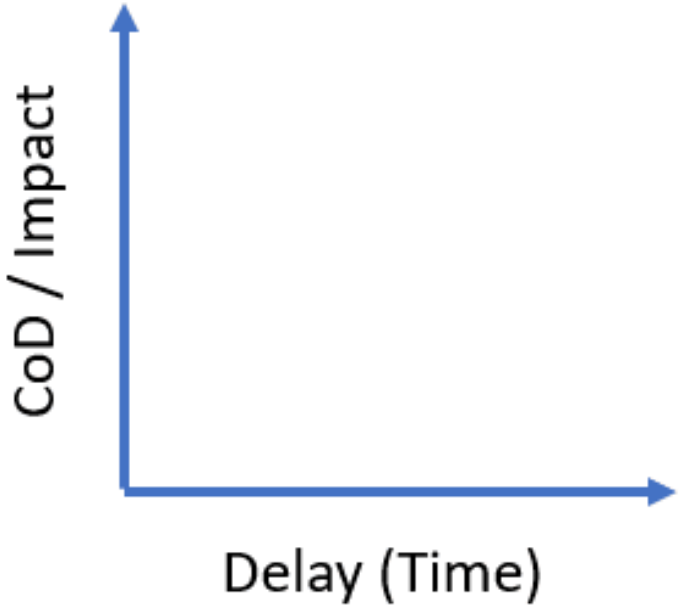
Lesson 4 – A need for Prioritisation

Many popular prioritisation frameworks, some included below…



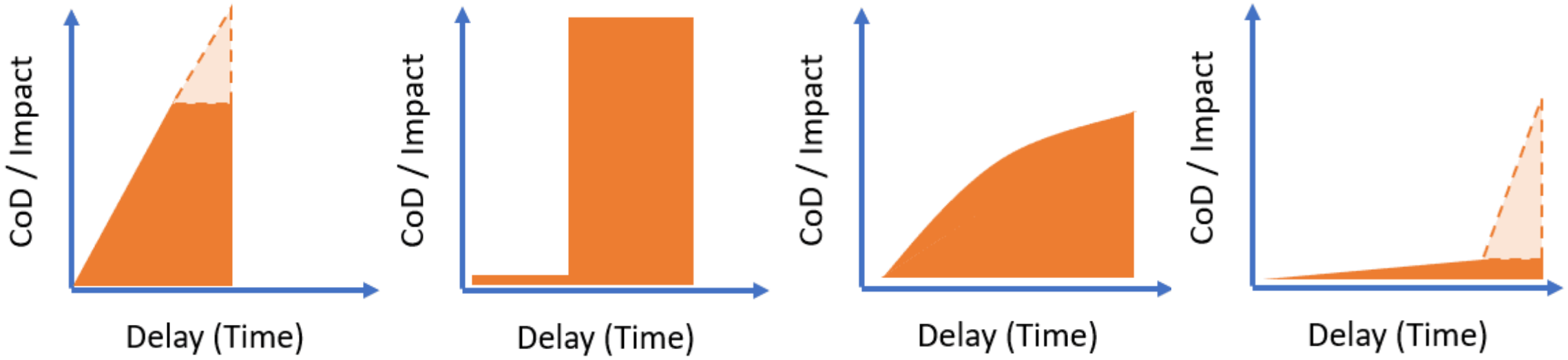
Lesson 4 – A need for Prioritisation

- The CoS (Classes of Service) technique uses the **Cost of Delay (CoD) prioritisation framework**
- To a team/organization – CoD is a principle that helps evaluate impact based on delaying a task, i.e. “Cost of Delay”



Lesson 4 – A need for Prioritisation

Relationship of the 4 x "Classes of Service" with "Cost of Delay"



Expedite

Fixed-Date

Standard

Intangible

(Majority Tasks)

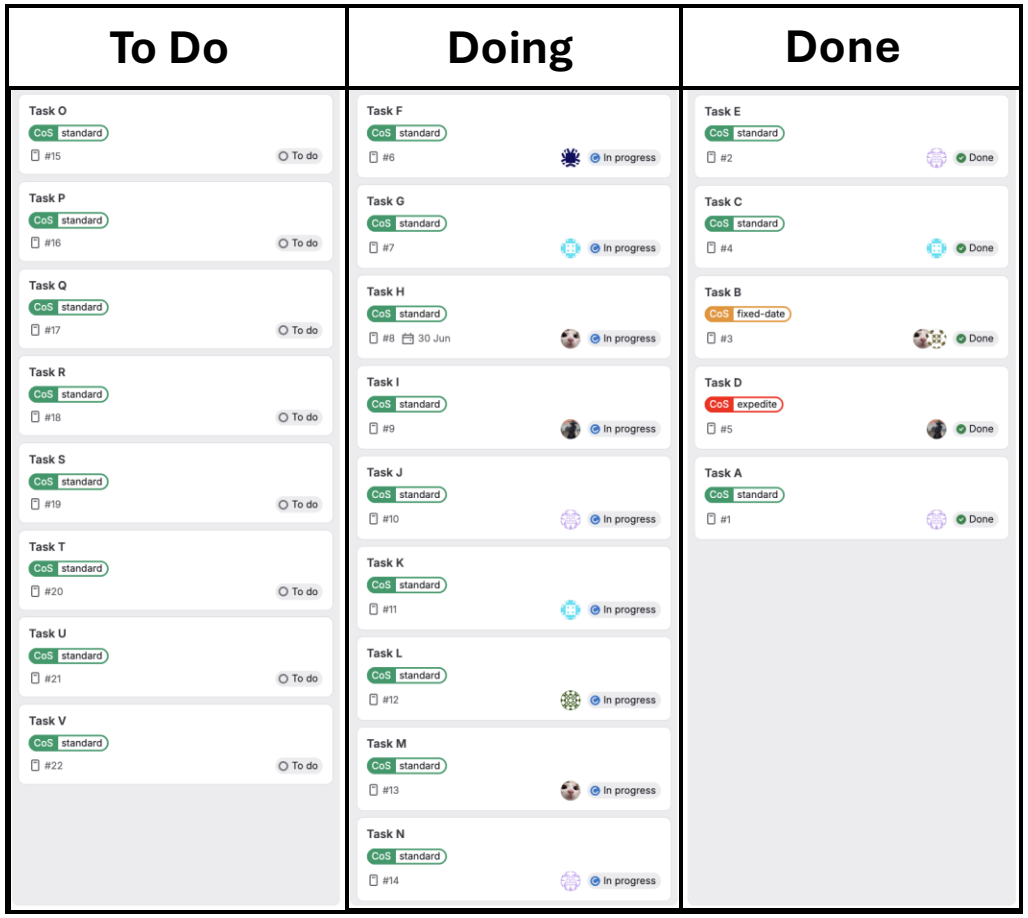
Ref: https://teamhood.com/kanban-resources/kanban-classes-of-service/

Lesson 4 – A need for Prioritisation

- CoD (Cost of Delay) is ultimately a good way to manage “risk”
- “Cost” does not only mean financial
- Risks come in many forms, some examples include:
  - Financial
  - Cyber Security
  - Service Availability
  - Reputational
  - Customer Satisfaction

Lesson 5 – Limiting Work in Progress (“WIP”) - a silver bullet!

- One of the fundamentals about the Kanban process is to limit WIP (Work In Progress) so individuals are not overwhelmed / distracted
- Ensured no individual was working on > 3-5 work items at once
- However, if these are not “shared” work-items, but 5 separate work-items each, this increased the overall “Team WIP”
- Became a real bottleneck for us!



Lesson 5 – Limiting Work in Progress (“WIP”) - a silver bullet!

- Idea was to limit WIP at ”team level”, not only for individuals. Started putting an arbitrary value - not working very well - i.e. not seeing great "**flow**" of work-items

- Experimented, researched and continously fine-tuned. More success with limiting WIP in "Doing" column

Q1 - How does a team know how well they are doing, i.e. how to measure progress?)

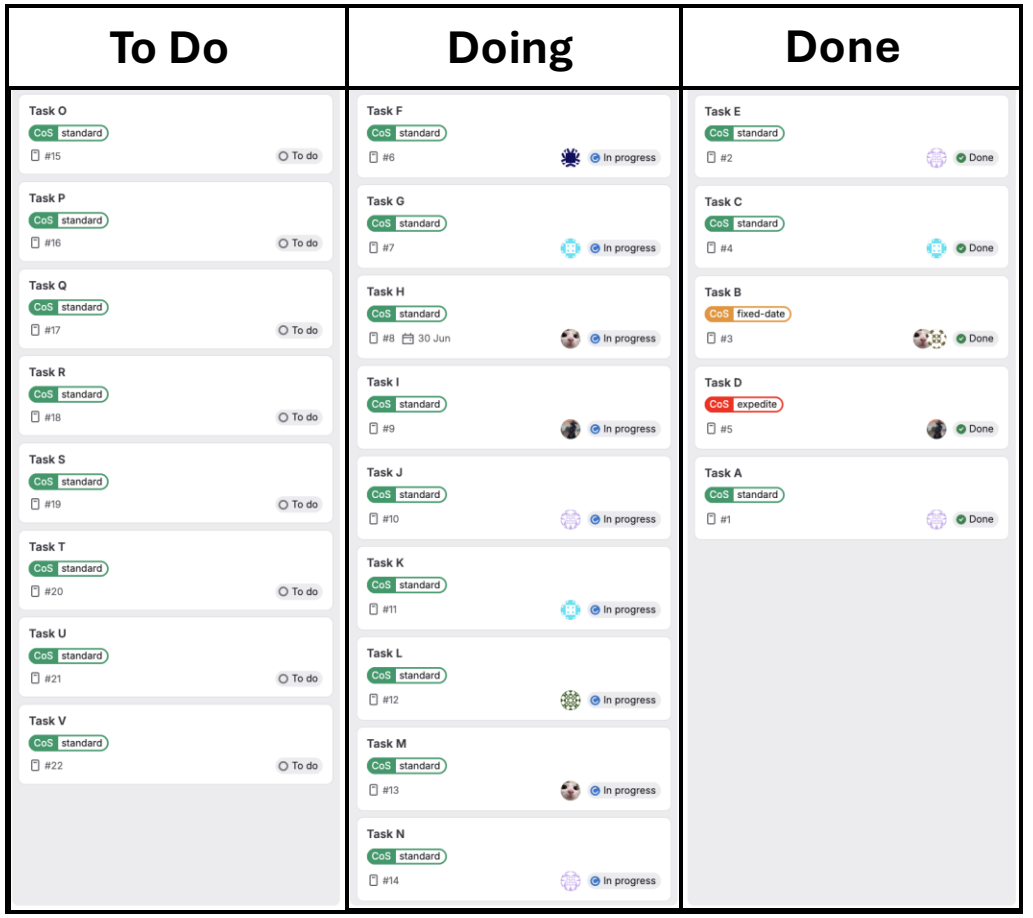
- Kanban is a method of managing workflow – “flow” being operative
- Focus of Kanban is to improve “flow” of tasks across the stages/columns, usually more columns than 3
- Some key “Flow Metrics” are:
  - Lead time
  - Cycle time
  - Work-in-Progress
  - Throughput
  - Flow Efficiency
  - Work-Item Age
  - CFD (Cumulative Flow Diagram)
  - Burn-down chart

Q1 - How does a team know how well they are doing, i.e. how to measure progress?)

- If you were to pick a couple metrics to start with:
  - Cycle time – amount of time a work-item spends “**In Progress**”
  - Lead time – amount of time a work-item spends from “**Request or Commitment**” to “**Resolution**”
- Philosophy – This is not a “prescribed” period – you measure what this is for your team/context and keep improving

Q2 - What "WIP" (Work In Progress) limit to use?

- Was a struggle at first
- Tried recommended WIP of n+2 (n= teamsize). This was not a bad call, but it depends
- Another suggestion mentions WIP limit is between 1n and 2n
- For a team of 10, WIP limit could be 10-20
- Likely in the middle is our experience so far, however very dependent on working environment
- Set something resonable and track what works – better answer over time



Q3 - How would work typically be broken down?

Q3 - How would work typically be broken down?

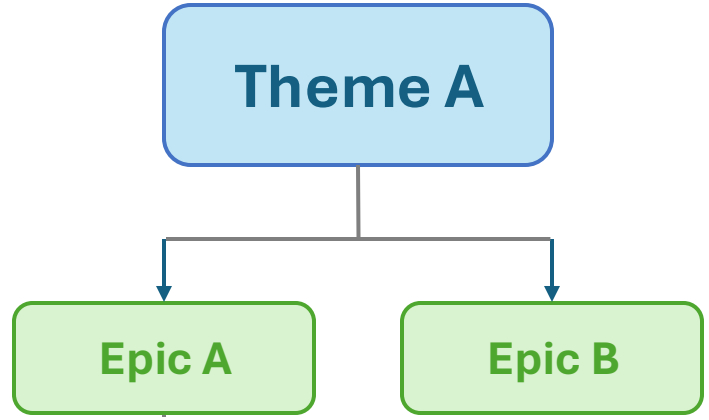
Initiatives/Goals 

Theme A

Q3 - How would work typically be broken down?

Initiatives/Goals ⇨

Projects ⇨

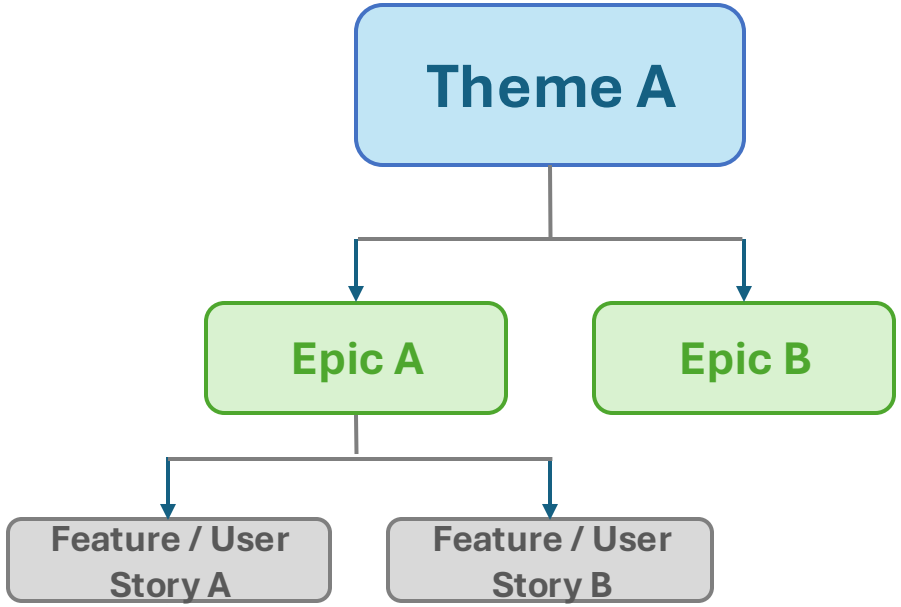


Q3 - How would work typically be broken down?

Initiatives/Goals ⇨

Projects ⇨

Functionalities ⇨



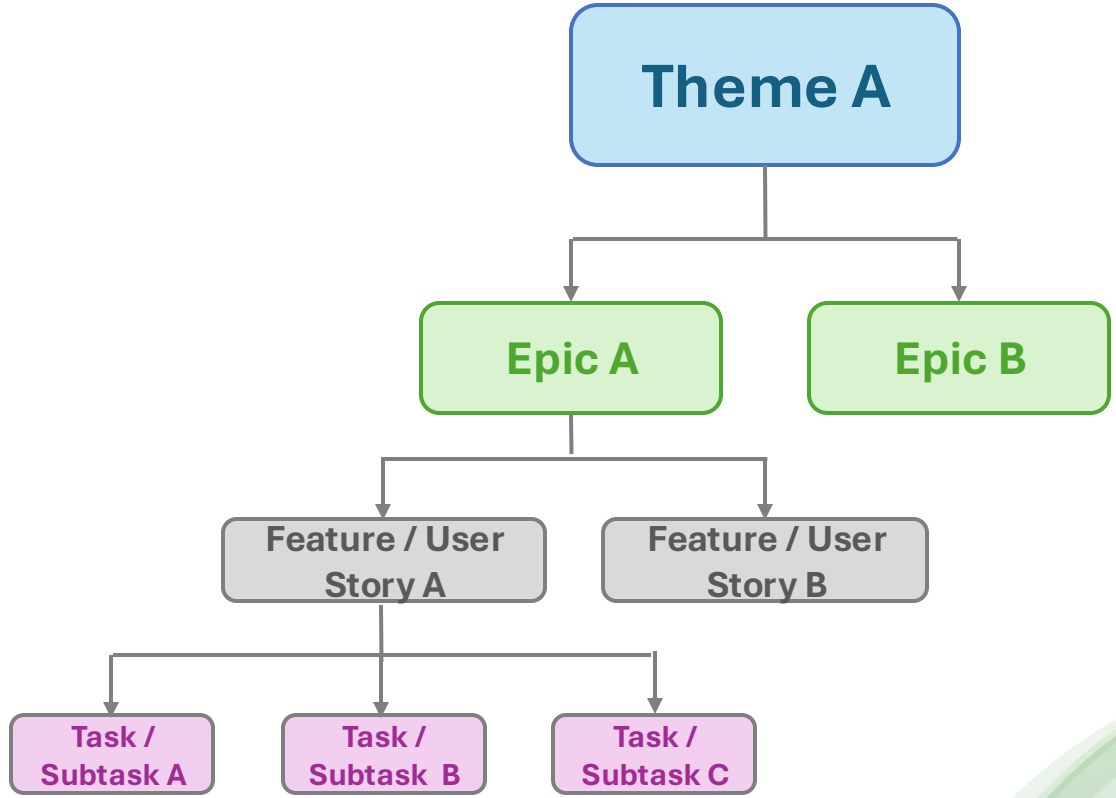
Q3 - How would work typically be broken down?

Initiatives/Goals →

Projects →

Functionalities →

Tasks →



Example - my card for today's talk...

[20th May 2026]

ACIT Hub - Prep Agile talk

CoS fixed-date

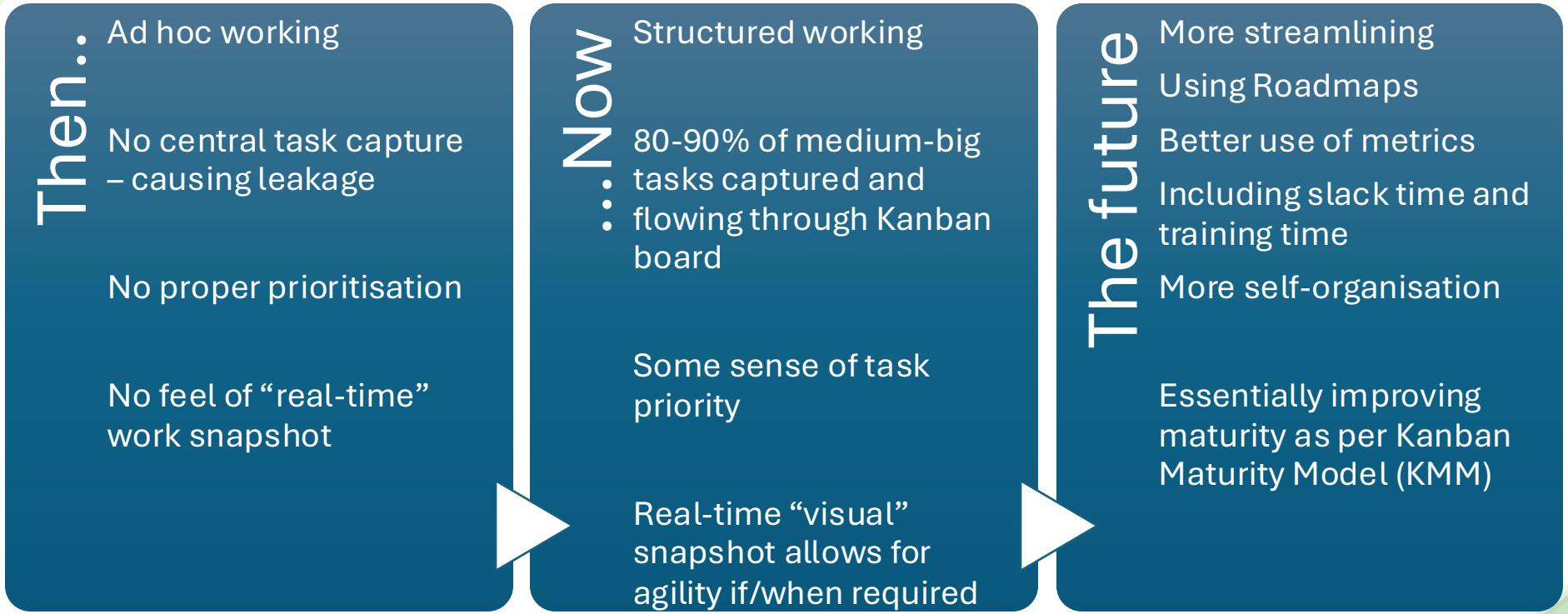
JM In progress

A People-Centred Approach

- Design processes to work for human beings, not against! **"Psychological safety"** – very important. Work transparency can feel revealing to some at first, however becomes more normal with coaching/time/understanding
- After all – process exists to expose bottlenecks and improve team – embrace it and approach with a healthy/positive attitude
- Agile will not solve all problems – good training, collaboration, leadership, thinking, documentation, initiative, culture, etc., still relevant
- Team-work – cards can be assigned to individual or small group – focus should be on how fast "team" is !
- Implementation
  - Start simple and increment slowly (be patient)
  - Use problems/frustrations as "signals" for next iteration
  - Need time for members to align with new processes
  - Need time to test if changes have worked

People dictate the boards and not the other way round !

Our learning Journey



ACIT Hub

Q & A

UNIVERSITY OF SURREY