Exercise – Sources of Delay (Part 1 of 2)

10 min

- Gather 5-10 sticky notes: Where in your organization does work wait (queue), and for how long?
- Cluster these sticky notes based on the source / cause for the delay! Do dependencies cause delay? Which type? Internal to your service, external to your service but internal to your business, external to your business?
- Is the length of the delay predictable or not? What is a typical lead time distribution?

Kanban University

Delay

Exercise – Sources of Delay (Part 2 of 2)

10 min

- On more sticky notes, write down any actions available to avoid or influence these different delays.
- Does avoiding delay change the risk in delivering the work?
- Can you imagine reducing the size of queues and length of queuing time? If so, what are the implications?
- Can you imagine circumstances where avoiding dependencies that cause delay would be acceptable?

Kanban University

Upstream

- Requirements Gathering BA teams Weeks Unpredictable
- Solutions Architecture Months Unpredictable
- Approvals Legal 2 Weeks Unpredictable

Delivery

- Core Systems Dependency Weeks Unpredictable
- Design Team Weeks Predictable
- QA bottlenecks (unpredic table)
- Ext Marketing agencies (upstream, predictable)
- Customer Contact (unpredictable)

Downstream

- RBNZ Review Months Unpredictable
- Cloud Team Weeks Unpredictable

Actions

- Transparency Cadences WIP
- Set up as a shared service
- Introduce Kanban/ Shared Service
- Automating Testing
- Dual Track Development
- Avoiding delay (sidestepping) will increase risk in delivery
- "reducing" delay will reduce risk
- Reducing queues and qt will improve flow and predictability
- Sometimes you have no choice - RBNZ.
- Bureaucracy happens - hoop jumping

- TRB ahead of UAT deployment
- "shift left", plus DevOps
- CAB ahead of production deployment
- "shift left", plus DevOps evntually
- Documentation signoff, architectural review
- Documentation as-code and review of delta

- Customer enagement
- Customer replenishment cadence
- vendor engagement
- More active management: transparency, cadenance

Tanya & Damon

Sources of Delay

Exercise – Dependencies (1 of 2)

10 min

1. On what service(s) do you depend, i.e. which ones do you call?
2. How do you identify in good time which work items will have dependencies on external services?
3. Can you predict when you will demand capacity on dependent services?
4. Where are you dependent on your customer(s)?

Kanban University

Dependencies

Exercise – Dependencies (2 of 2)

10 min

Walk in the dependent service(s)' shoes!

- How does demand look from that perspective?
  - How do they manage demand that is blocking or waiting?
  - Do they have multiple sources of demand?
  - Do they offer different classes of service?

Kanban University

CLIF Improvements

- NZLT
- Talent & Culture
- Finance
- No
- Visualise the activities & create a workflow
- Regular Cadences to manage flow
- Early signals on tickets
- Everyone is a customer when it comes to improvement

- NZLT - Variable - comes in as needed They can request at any time. YES YES
- Talent & Culture - Variable - comes in as needed No demand foreseen YES YES
- Finance - Variable - comes in as needed No demand foreseen YES YES

- Damon & Tanya
- 1. Platform Architects
- 2. Quarterly Product roadmaps, release planning, learnings all self highlight when they are needed
- 3. Yes we can predict
- 4. Upstream product team engage with customers, customer relationship management but the delivery team does not with exception of impact from PO when needed.

- not sure, they have their own priorities, perhaps the hippos
- yes approx. 4-5 teams each
- yes we have several classes now and bigger teams get priority, but we are moving to a standardised set of service classes all services

- Architecture Service
- Upstream Kanban
- Unpredictable, ad hoc Forecast
- Multiple Sources Yes Services It looks a Kanban Board with cadence
- Ability to absorb change (TEC)
- Subject Matter Expert UAT
- Embedded into team
- Planned, Sprint and phasing
- It looks important
- Customer view / sign off
- PWS Service development team sythesis and analysis
- Shifting to a Kanban Board
- Unpredictable, ad hoc
- Multiple Sources Work in progess
- Other programmes of work
- ?
- It's not working
- Multiple sources

Dependencies

Bottlenecks in day-to-day life

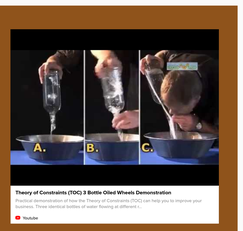
10 min

- As a group, take 2 minutes to collect as many "bottleneck situations" you know from your day-to-day life as you can. Put them as stickies on the virtual whiteboard.
- Select some of the examples, and add information about:
  - The cause of the bottleneck
  - The impact on you
  - How it has been dealt with / How it could have been dealt with.

Kanban University

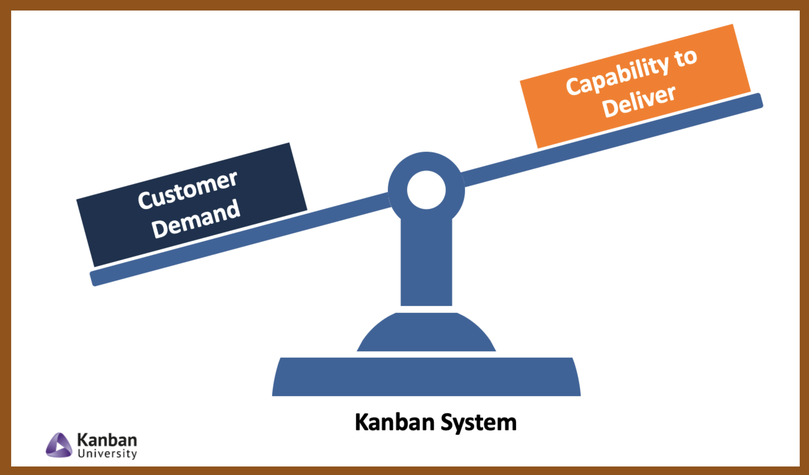
Bottlenecks

- Waiting for my train this morning
- waiting to board your flight
- the bus is full and it passes me by without stopping
- queue at the supermarket & any other queue
- waiting for the lift
- Weather stops the ferries
- Costco on a Saturday
- El Nino and Water Delivery
- Foodbag Bottleneck when you don't make all the meals you ordered .... compounding over weeks
- Learning songs in my band
- Prepping house for sale
- Contractor sychronisation on my rennovation
- Washing / drying with only one five electical outlet
- Have to unplug the dehumidifier to use the hairdryer
- Roundabouts at peak hour
- Disc space on my laptop for astrophotography
- Lack of Sun - Outdoor jobs build up in Auckland



Bottlenecks

What improvements can you think of that can shape the demand or improve delivery capability?



Demand and Capability

The Five Focusing Steps

10 min

- In your groups, select a bottleneck situation from private or work lives.
- What is the typical response to this situation?
- Now, apply the 5 focusing steps to it.
- What insights did you gain? Did you discover other options you did not think about at first?

Kanban University

Five focusing Steps

- The Five Focusing Steps:
  - Identify the system's constraint(s).
  - Decide how to exploit the system's constraint(s).
  - Subordinate everything else to the above decision(s).
  - Elevate the system's constraint(s).
  - Start all over!

Identify

- Waiting for Testing

Typical Responses

- Hire More Testers
- Delay/ Accept the futility of life
- Offshoring/ Outsourcing
- Release without Testing

- RCA
- Testing Capacity
- Too much demand
- Disproportionate ratio of Engineers/ Testers

Exploit- use your existing resources/ look internally

- Shift the burden left
- Split the Stories
- Which stories should be started to reduce redundancies in testing
- Developers swarm to help testers
- buffers & WIP limits
- TDD

- Subordinate- coordinate, collaborate, cultivate/ look externally
- resources from other teams helping
- outsourcing

Elevate- measures that cost you the most

- Hire more testers
- more automation
- Test Automation
- Introduce TDD

Five Focusing Steps