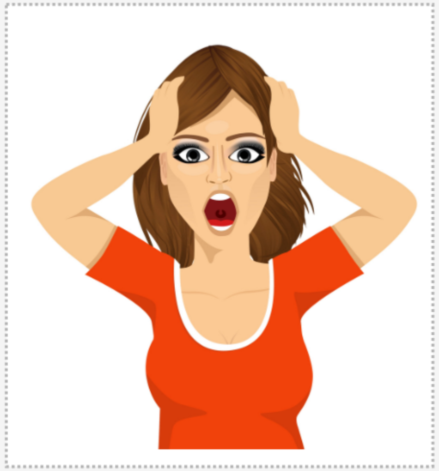
Customer-Driven Value Streams

November 2022



Tammy the trade-in Specialist

"Wants to reduce customer abandonment in the cell phone trade in process"

Who is this person?

Been with a telecommunications company for 20 yrs

Lives in New Jersey

She is reponsible for the cell phone trade-ins program at her company

Is also responsible for 6 other products

What Hurts?

Existing Customer Trade-ins process is cumbersome

Customers often abandon trade-ins midway

Too many meetings

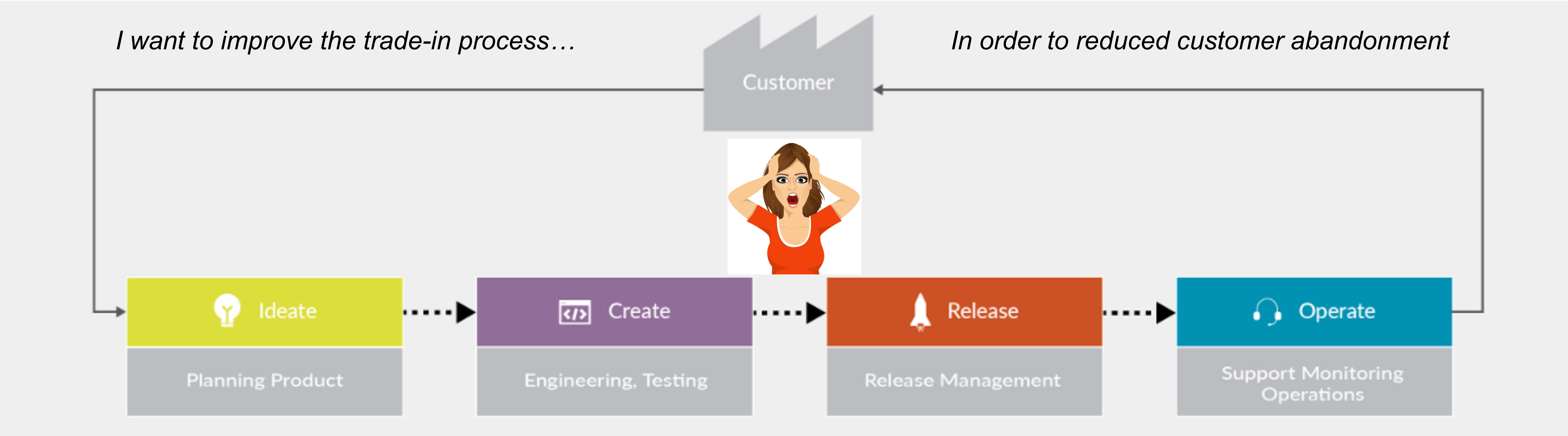
Has difficulty focusing due to workload

What do your customers care about?

What do your customers NOT care about?

Value Streams

What is a product value stream (and Why)?


I want to improve the trade-in process…
In order to reduced customer abandonment
Customer
Ideate
Planning Product
Create
Engineering, Testing
Release
Release Management
Operate
Support Monitoring Operations

ALL the Activities →

POC
Wireframe
Business Case
Story Grooming

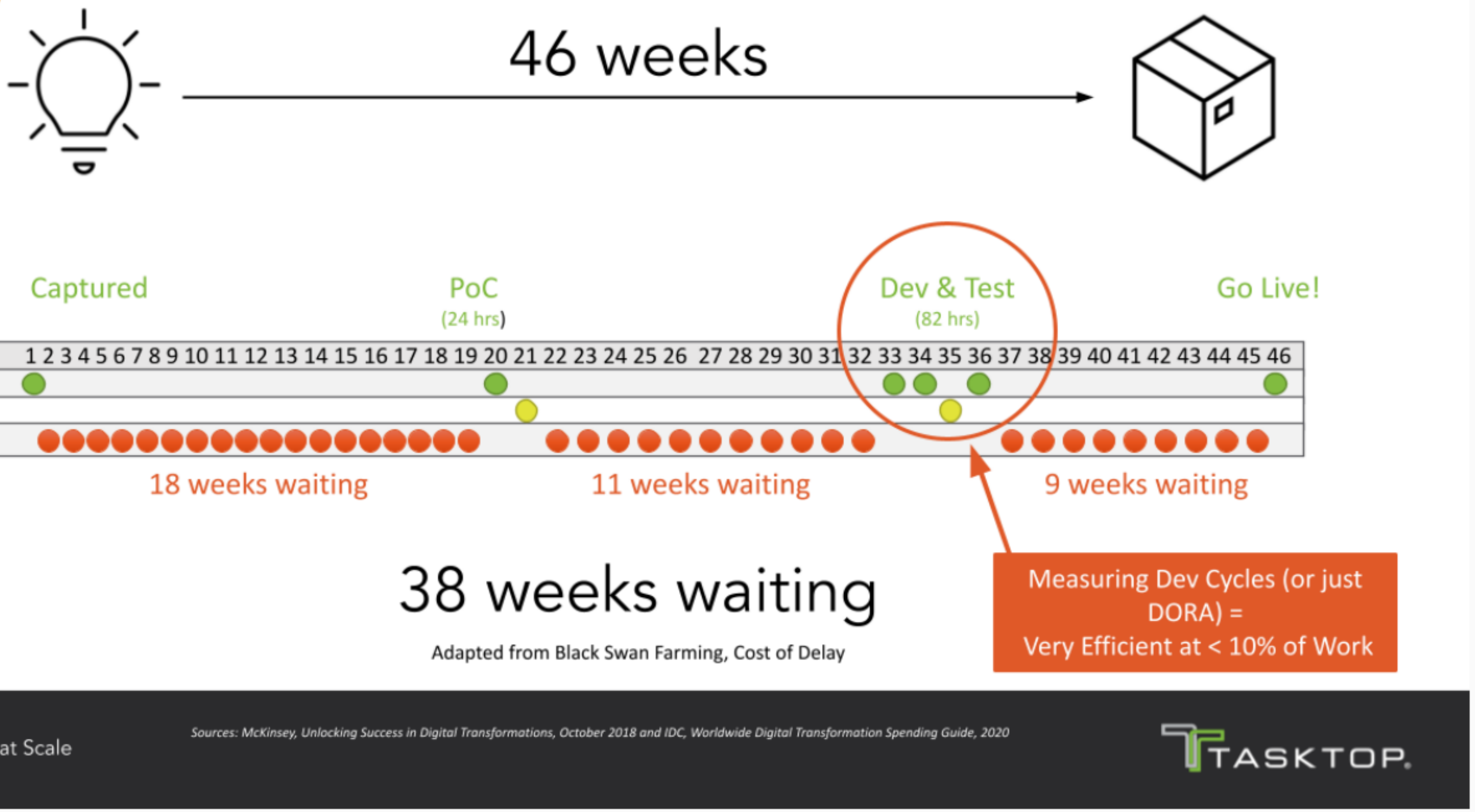
Coding
Testing
Security Remediation
Refactoring

User Acceptance
Regression Testing
Change Management
Production Deployment

Production Defects
User Support

The concept of a value stream is aligned to the customers point of view

What is the customer's point of view?


46 weeks
Captured
PoC
(24 hrs)
Dev & Test
(82 hrs)
Go Live!
Week 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46
Value Add
Risk Reduction
Wait Waist
18 weeks waiting
11 weeks waiting
9 weeks waiting

38 weeks waiting

Adapted from Black Swan Farming, Cost of Delay

Measuring Dev Cycles (or just DORA) =
Very Efficient at < 10% of Work

Master Software at Scale
© 2021 Tasktop

Sources: McKinsey, Unlocking Success in Digital Transformations, October 2018 and IDC, Worldwide Digital Transformation Spending Guide, 2020

TASKTOP.

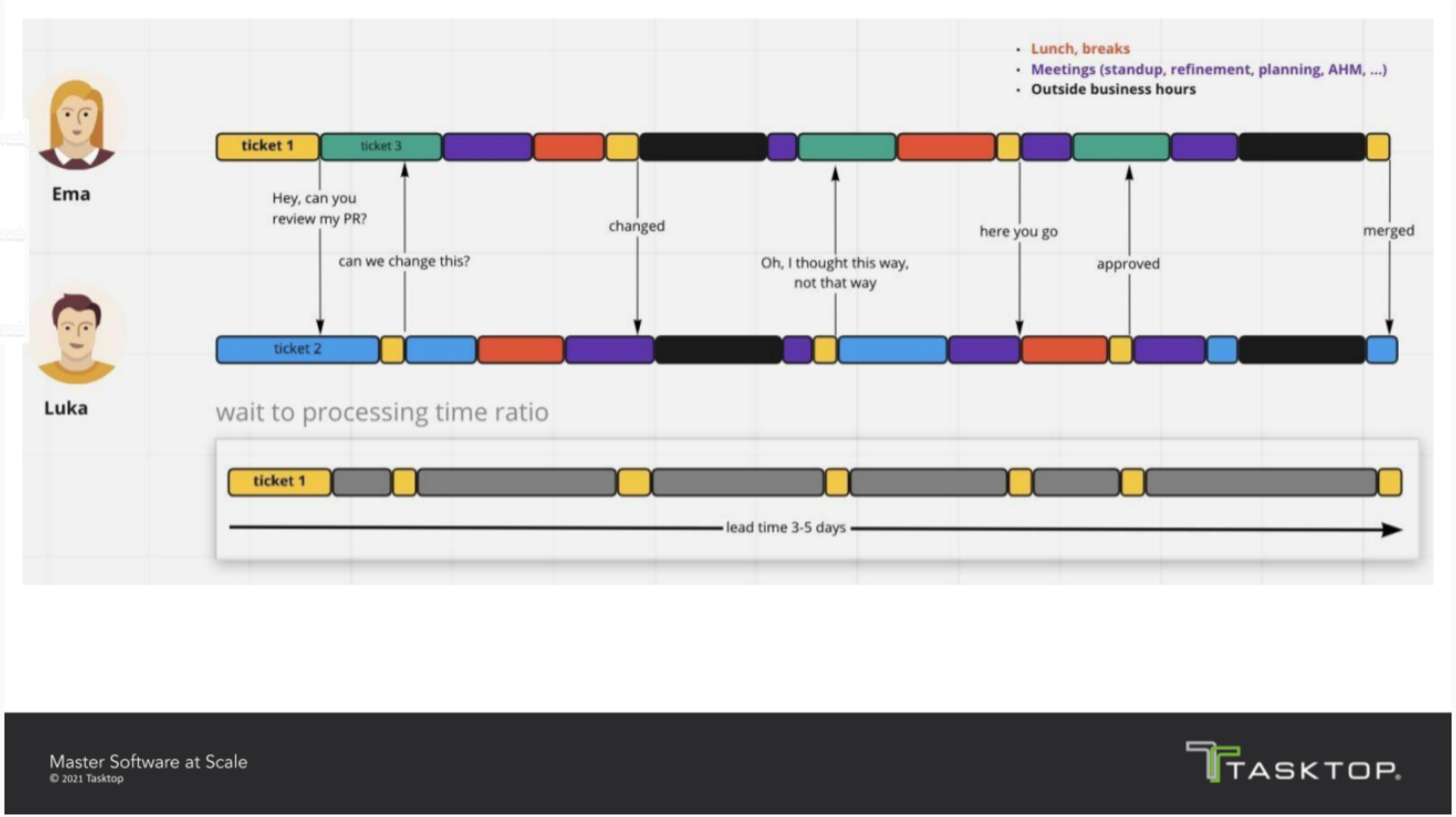
5 days ☺

No idea ☹

Manager

What is the customer’s point of view?

5 mins ☺



- Lunch, breaks
- Meetings (standup, refinement, planning, AHM, ...)
- Outside business hours

Ema

ticket 1

ticket 3

Hey, can you review my PR?

can we change this?

changed

Oh, I thought this way, not that way

here you go

approved

merged

Luka

ticket 2

wait to processing time ratio

ticket 1

lead time 3-5 days

Master Software at Scale
© 2021 Tasktop

TASKTOP.

What defect?

Manager

5 days

Demand & Capacity

Understanding Demand

I want reduce customer abandonment

VALUE DEMAND

What is Tammy asking for?

+

I need to address these security vulnerabilities

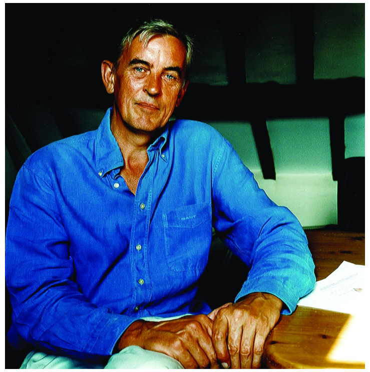
This button doesn’t work!

I need to rewrite that class

FAILURE DEMAND

What is really taking all our time and focus?

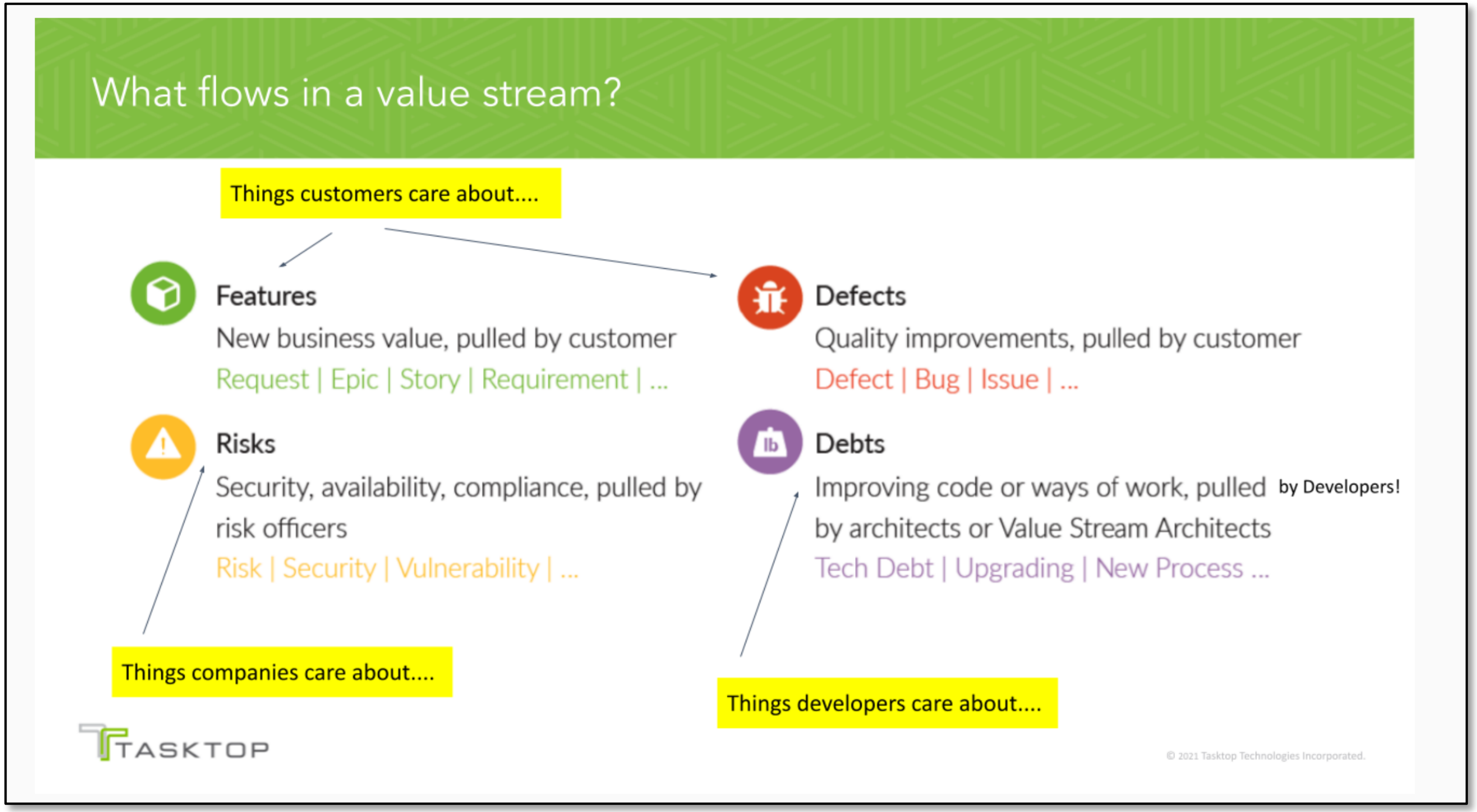
=



SYSTEM CAPACITY

What percentage of your team’s work is failure demand?

Flow Framework Flow Types



What flows in a value stream?

Things customers care about....

Features

New business value, pulled by customer

Request | Epic | Story | Requirement | ...

Defects

Quality improvements, pulled by customer

Defect | Bug | Issue | ...

Risks

Security, availability, compliance, pulled by risk officers

Risk | Security | Vulnerability | ...

Debts

Improving code or ways of work, pulled by Developers! by architects or Value Stream Architects

Tech Debt | Upgrading | New Process ...

Things companies care about....

Things developers care about....

TASKTOP

What if we combined them?

Value Demand

Failure Demand

Features

Defects

Customers traditionally don't see below the line

We want to elevate the discussion about risk and debt in order to properly set expectations and improve flow of value

Risk

Debt

I want to fix debt

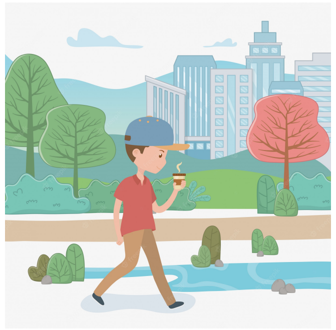
Capacity

Customers don’t see our capacity, but they sure feel the impacts if demand and capacity are not properly balanced

Capacity – FIFO Queue

Arrival Rate

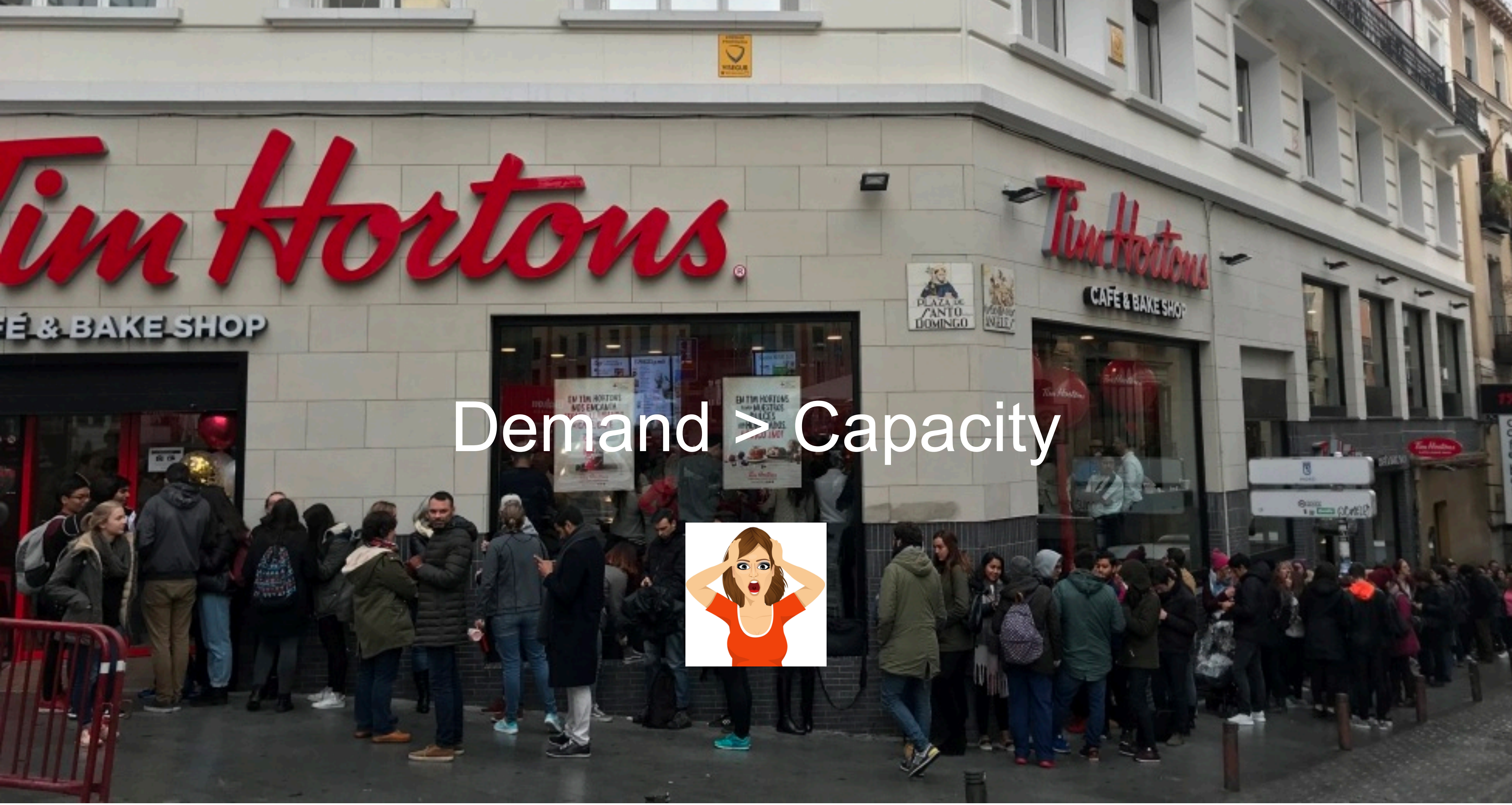
Service Time



Departure Rate

Demand

Capacity



Demand > Capacity

Capacity – Prioritized Queue

Arrival Rate

Service Time

Departure Rate

Demand

Capacity

Demand > Capacity

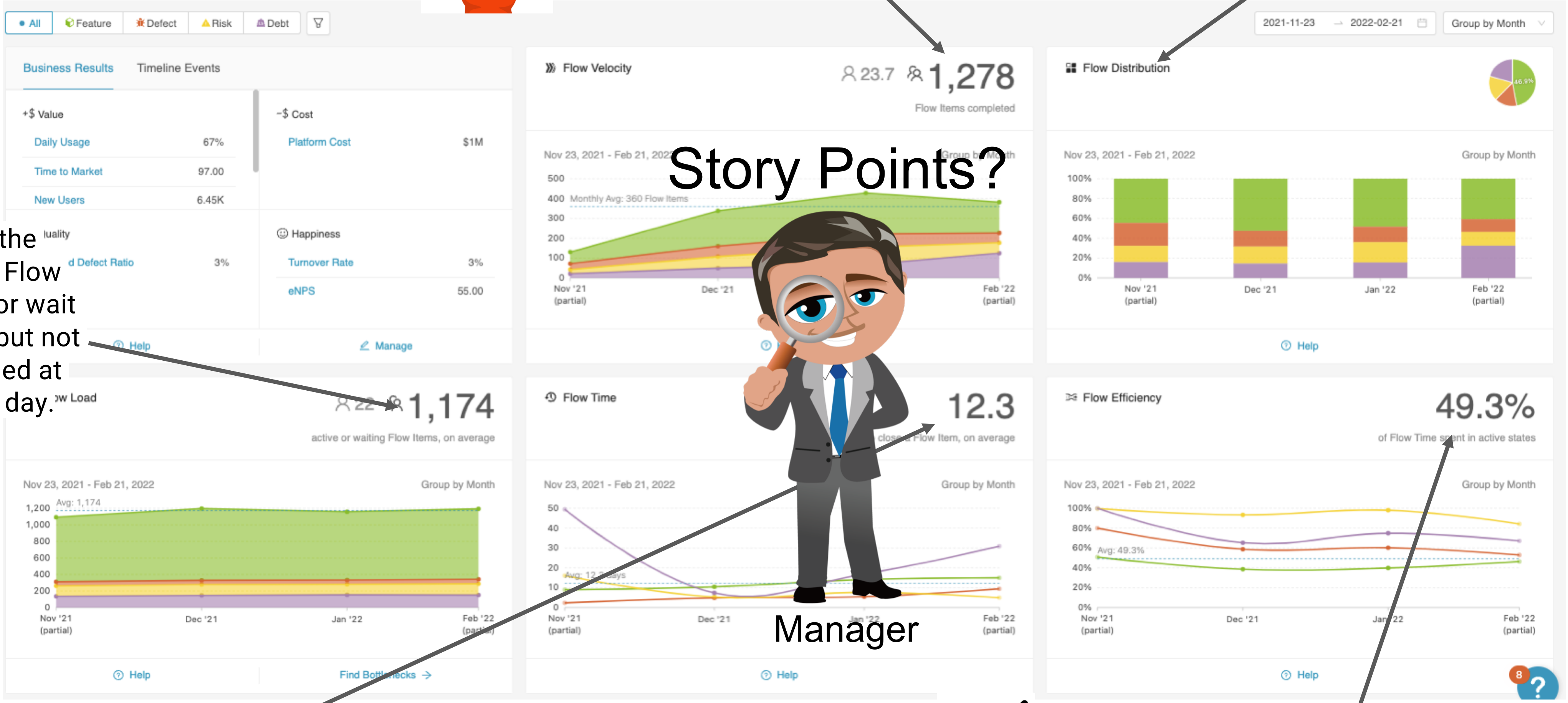
Measuring Flow

Measuring Flow

Flow Velocity is defined as the number of completed Flow Items minus the number of re-opened Flow Items per time interval.

Flow Distribution is the relative distribution of Flow Velocity across the four Flow Item types (feature, defect, risk, debt).

Flow Load is represented as the total number of Flow Items in active or wait states (started but not finished) recorded at the end of each day.



Business Results | Timeline Events

+$ Value

Daily Usage	67%
Time to Market	97.00
New Users	6.45K

-$ Cost

Platform Cost	$1M

uality

d Defect Ratio	3%

Happiness

Turnover Rate	3%
eNPS	55.00

Flow Velocity — 23.7 — 1,278 Flow Items completed

Flow Distribution

ow Load — 22 — 1,174 active or waiting Flow Items, on average

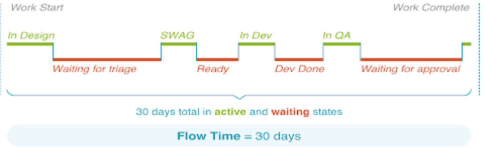
Flow Time — 12.3 close a Flow Item, on average

Flow Efficiency — 49.3% of Flow Time spent in active states

Story Points?

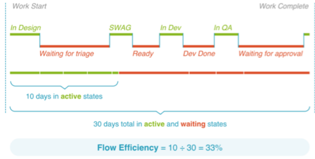
Manager

Flow Time is calculated by averaging the total days work remains in both active and waiting states from work start to work complete.



Flow Time = 30 days

Flow Efficiency is calculated by dividing the number of days a Flow Item stays in an active state by the number of days a Flow Item stays in active and waiting states combined.

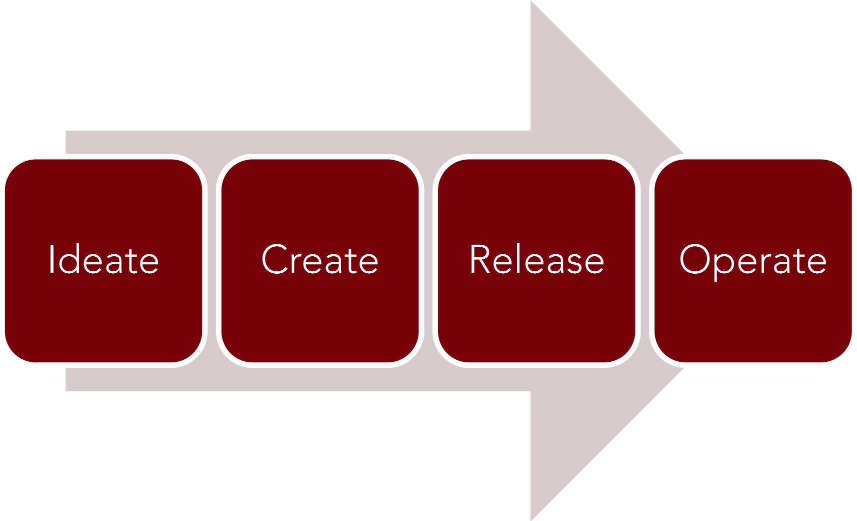


Flow Efficiency = 10 ÷ 30 = 33%

Visualizing Flow

Value streams are often not linear

What we think it looks like…



Ideate

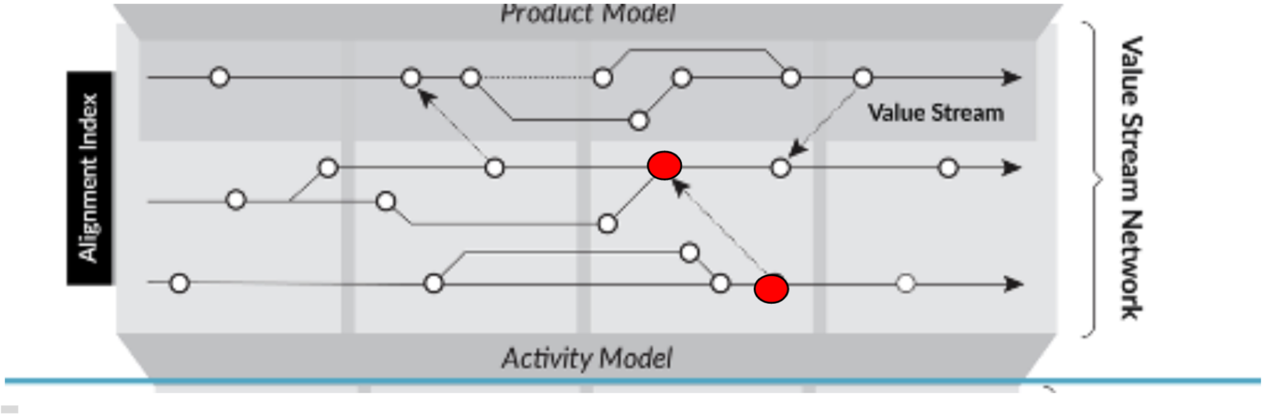
Create

Release

Operate

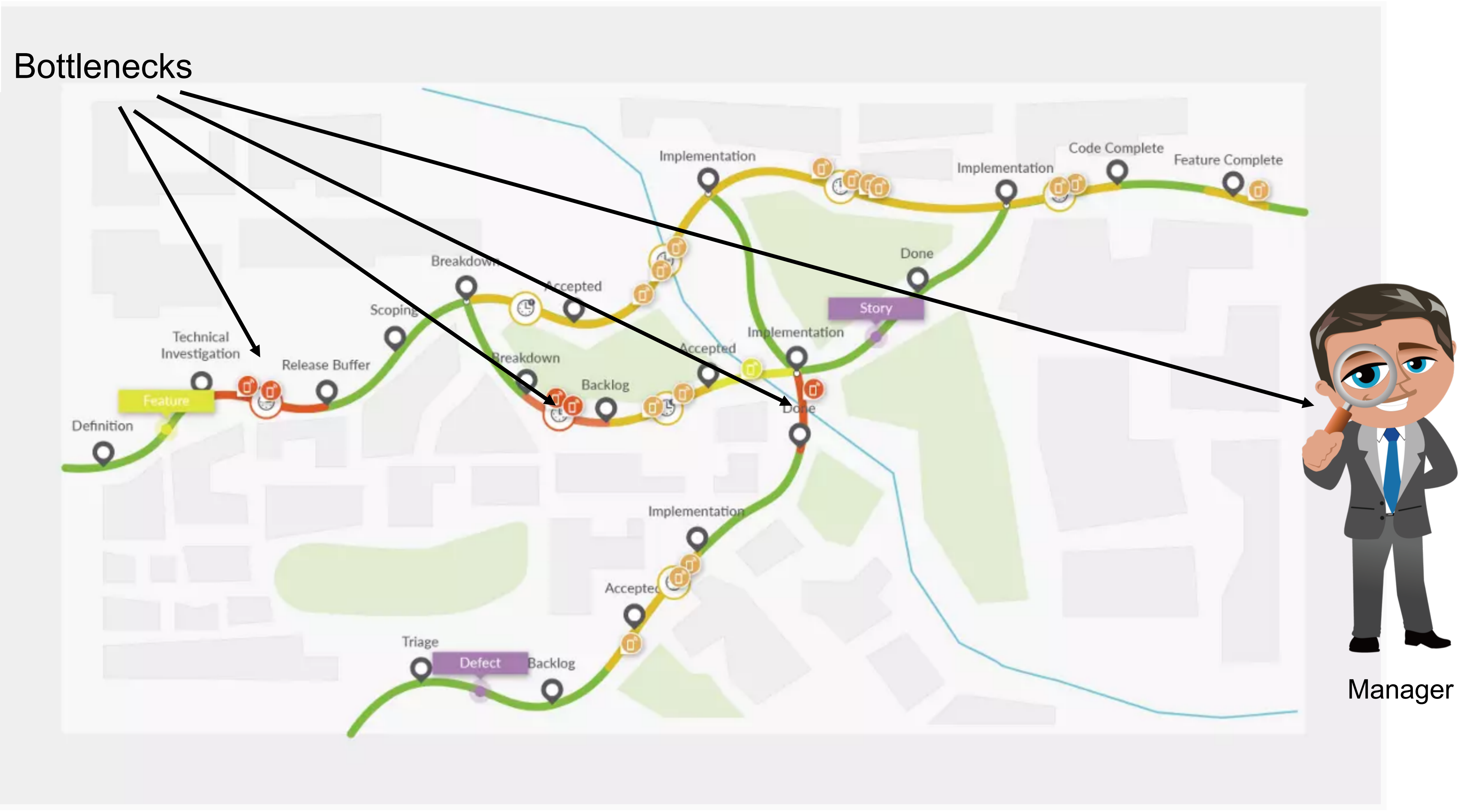
Value does indeed flow left to right…

It looks more like a map of interconnected systems



"The way work flows through an organization's Value Stream Network is one of the least understood components of value delivery. Because this network was evolved entropically rather than intentionally, it's both nebulous and pervasive which makes understanding the flow of value delivery very challenging. One cannot manage or improve what one cannot visualize and measure."   -Carmen Deardo

Visualizing Flow

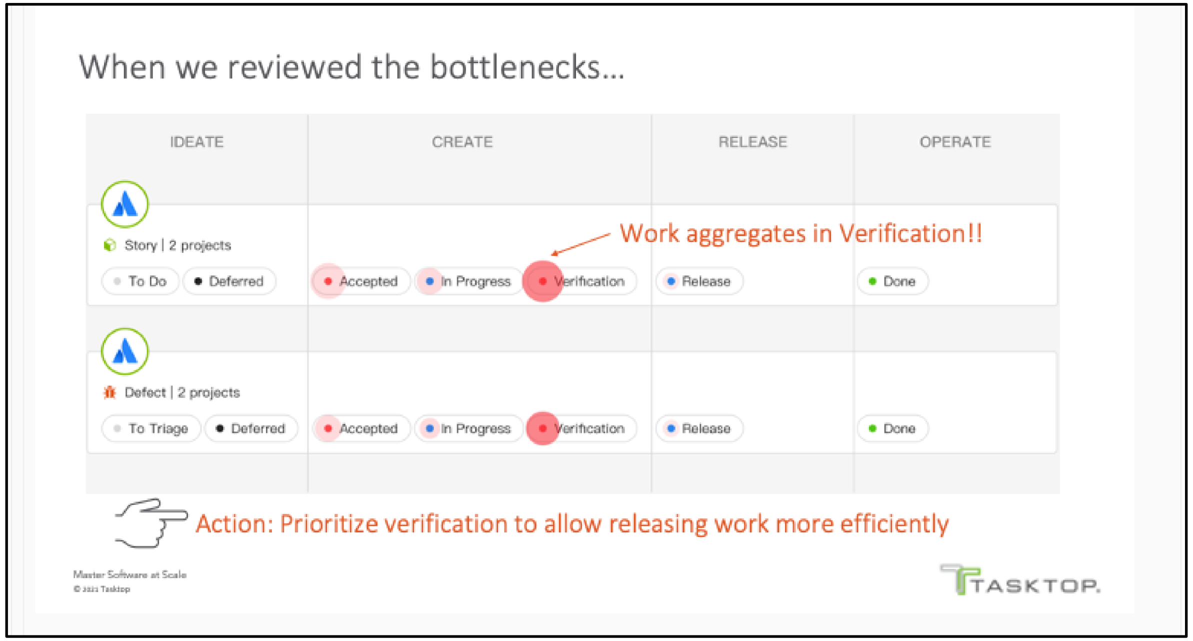


Bottlenecks

Manager

Improving Flow

Flow Experiments



When we reviewed the bottlenecks…

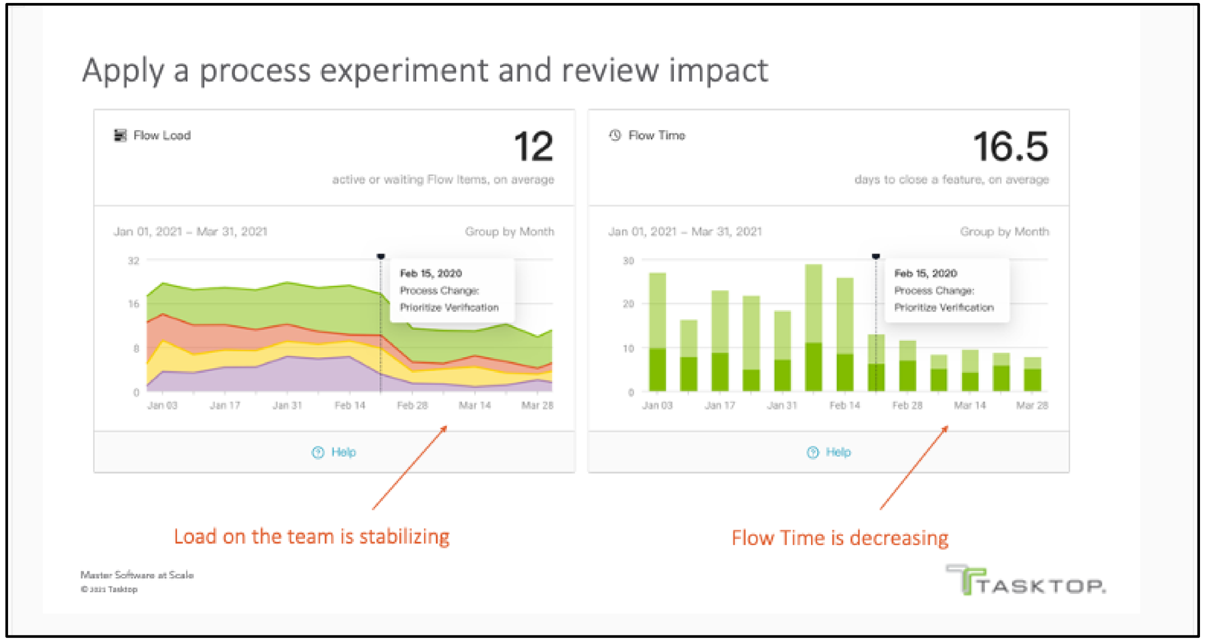
IDEATE	CREATE	RELEASE	OPERATE
Story | 2 projects ● To Do ● Deferred	● Accepted ● In Progress ● Verification	● Release	● Done
Defect | 2 projects ● To Triage ● Deferred	● Accepted ● In Progress ● Verification	● Release	● Done

Work aggregates in Verification!!

Action: Prioritize verification to allow releasing work more efficiently

Master Software at Scale
© 2021 Tasktop

TASKTOP.



Apply a process experiment and review impact

Load on the team is stabilizing

Flow Time is decreasing

Master Software at Scale
© 2021 Tasktop

TASKTOP.

Tony

Experiment Title: *Get home by 5 pm*

Background

- Tony’s team was only allowed to release code to prod every 2 weeks after hours
- This created a situation where IT & the business was staying until 9 pm every other Monday

Current Situation

- Due to some recent poor release quality, the business had grown uncomfortable releasing code frequently. This led to bi-weekly releases.
- The release process was manual

Goal

- Get everyone home by 5 pm
- Improve flow & automate the release

Analysis

- The team was fully capable of releasing weekly
- The bi-weekly batch size was leading to higher risk of defects
- The business needed more confidence in IT

Countermeasures

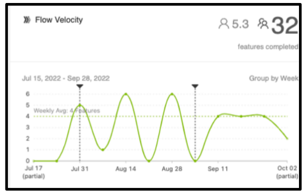
- Do nothing and risk losing people
- Investigate alternative deployment windows

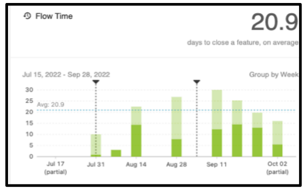
Action Plan

- Experiment with weekly deployments
- Lower batch size
- Automate deployment process

Results

- Moved from big batch to frequent flow of value. Flow time dropped.





- Employee happiness went up
- Release process was automated 100%
- Home by 4:30 pm ☺

Jacki

Experiment Title: *Stop piling on*

Background

- Jacki’s business team was only seeing about 40% of the planned features getting released
- Frustration was mounting in business and IT

Current Situation

- The business was prioritizing stories without awareness of IT capacity
- IT was afraid to tell the business they were struggling with completing the volume of requests.

Goal

- Improve flow and predictability for features
- Improve overall employee happiness for business and IT

Analysis

- The business team was grooming 2x the number of stories each month compared with the capacity of the IT team
- IT was accepting side door demand (much of it failure demand) that was consuming available capacity

Countermeasures

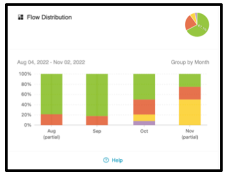
- Do nothing and risk losing business subscriptions

Action Plan

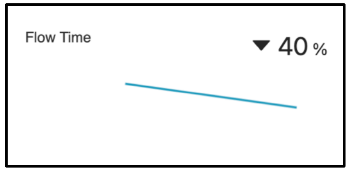
- Cut the number of stories being groomed in half
- Mature the prioritization process to include failure demand

Results

- Business began prioritizing features, defects, risk and debt (all the work)



- Feature flow time dropped by 40 %



- Greater trust and happiness amongst all stakeholders

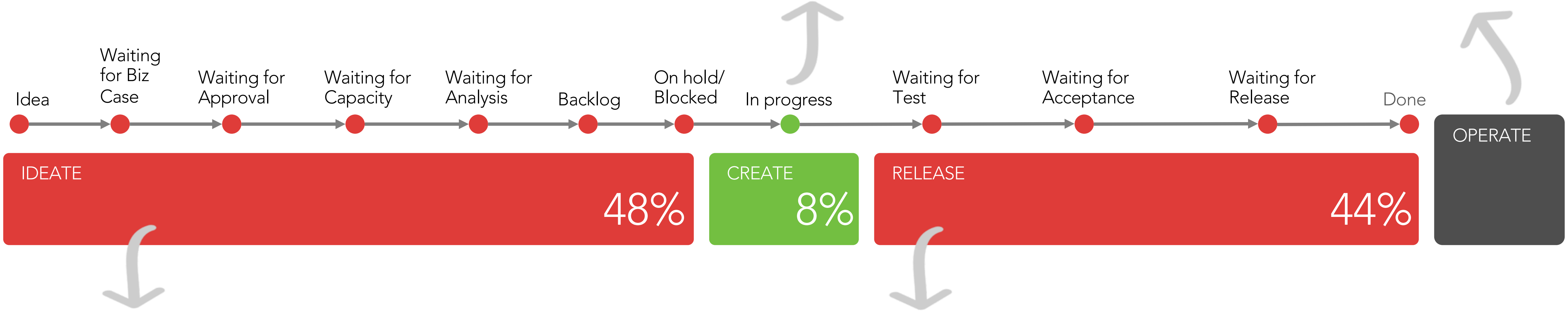
Questions???

Drill-down into software development and delivery

Many organizations miss the mark

"Agile" but not nimble
Relative acceleration during development

Poor quality creates unplanned work
Incidents and war rooms throw off plans



Heavy planning and funding processes
Starting and stopping from frequent cancellation, replanning and reprioritization

DevOps but not at scale
Delivery slowed down by dependencies,
tech debt and manual processes