

What is Operations? Ask the crowd...

- Sys Admin
- Network
- Application Bug Fix
- Procurement
- Datacenter Work
- Security

Operations is "Keeping the lights on!"

The Problem

- Ops is seen as "obstructionist" because they're often overworked
- Many times projects go over budget because of delays, lack of resourcing, etc.
- Behind schedule because of constantly shifting priorities, dealing with the latest "fire"
- We often never get to "done" because a project is "good enough" and we get yanked into different work.

Why?

- Capacity is not free time, we need time to plan, context switch, do administrative tasks, pee...
 - o Existing paradigms assume free second equals my work gets done
 - o Freeway analogy:
 - Capacity is not "free time"
 - When a freeway hits ~65% capacity, it starts to slow down
 - As we approach 100% capacity, everything stops
- Squeaky Wheel Problem
 - o Loudest complainer is not always the highest priority
 - o Why not reward those who PLAN?
 - o Better scheduling means more throughput (better capacity utilization), again let's talk about a freeway
 - Single Rider Cars vs. Bus
 - Pickup vs. Freight Train
- Humans suck at multitasking and context switching:
 - o Texting while driving
 - o Talking on the phone while reading a book
 - o Two Volunteers
 - Math problems

What is Agile?

- Agile manifesto
- Attempt to address these issues
- Focus on embracing change, small iterations, continuous improvement

Agile vs. Scrum

- Time Boxed vs State Constrained
- When do we reprioritize
- Defined Roles
- Team makeup
- Work item size/complexity
- Commitments

What is Kanban?

- Japanese, translates to Signboard (Name for cards place on bins)
- Developed by Taiichi Ohno, an industrial engineer for Toyota based on observations of supermarkets
- Implemented in the late 1940's
- Demand driven Inventory Control System
- A method for just in time production using Kanbans (cards)
- Limit Work In Process Inventory ← key
- Promotes Continuous Improvement
 - LEAN
 - Six Sigma
- Push vs. Pull Analogy (Supermarket)

Why Kanban for Ops?

- Time Boxing
 - Nobody ever gives us the whole picture (scope)
 - Interdependence on outsiders (facilities, vendors, other teams)
 - Stuff gets “back burner’d”
- Visualization
 - Accountability to the TEAM
 - Visualize blocked tasks
- Limits our WIP
 - Reduces context switching
 - Helps prevent burnout
 - Let's stakeholders set priorities
- Shifting priorities
 - Outages
 - Troubleshooting
 - Security
 - “Stuff Happens”
- We Suck at Estimating
 - Scope Creep
 - Need outside help
 - Things don't always follow “The Plan”
- Metrics for Ops
 - Throughput Matters
 - Latency (Time to resolution)
 - Resource Utilization
 - Flexibility/Agility

- Quality (minimize surprises)

How?

- This is just one way
- Get a board
 - Some tape works fine
 - Avatars
 - Cards
 - Categories (Our Example)
 - Priority
 - Flags
 - Blocked
 - Urgency (Our Example)
 - Late
 - Swim Lanes
 - People (Our Example)
 - Types of Work
 - Priorities
- Define States
 - Three Simple States
 - Scheduled
 - In Progress
 - Done
 - Other Examples? Ask Crowd
 - Planning
 - Waiting on Customer
 - Pending Approval
 - Blocked
 - Test
 - Deployment
- Put our work in inventory
 - Tasks become the cards
 - Describes something we need done
 - Moves through the “board”
 - Exists in only one state
 - Should be as uniform as possible
 - Pointing
 - Unit of measure (hour, day, etc)
 - Estimate not a contract
 - We’ll suck at it to start with
 -
 - Instead of bins, we pull tasks from a “backlog”
 - We (or our manager) prioritize the backlog
 - Once an item leaves the backlog, shouldn’t change priority or points. We’ll fix that next time (continuous improvement)
- Limit Work in Progress

- Set limits low to start with
- Only states that don't have WIP limits are backlog and done
- Don't break WIP limits
- If a task is stuck, team members pitch in to help get it unstuck before pulling more work to avoid breaking WIP limits
- Focus on reducing latency (time for an item to get through the system)
- Rarely change limits
 - Make a deliberate team choice
 - Should be done to increase throughput
 - Lowering can help solve queue starvation where cards never move because too many items are in progress
- Don't start new tasks until we get the current WIP to "done"
- This is a PULL system, not a push system.
- Review, Revisit, Revise
 - Daily "standup" helps keep the team accountable to each other
 - Call for help, visualize blockers, etc
 - Should be quick
 - Each team member should answer three questions
 - What did I accomplish yesterday
 - What will I accomplish today
 - What do I need help with
 - Routinely revisit priority of backlog
 - While items are in backlog, we should revise estimates, deadlines, commitments with new information
- Focus on getting tasks to "done".
-

Remember, be flexible:

- Toyota had six rules
- We only have two
 - 1) Visualize your work
 - 2) Limit work in process (WIP)

Other Tools:

- JIRA
 - Commercial ticketing system
 - Agile module allows kanban or scrum boards
 - Integration with other Atlassian tools (Stash, Confluence, HipChat)
 - Awesome code repo integration, tie commits to tickets (cards)
 - Tickets can become "cards"
 - Templating, integration, triggers, time tracking, completely customizable workflows, lots of plugins, a giant pile of awesome sauce
- Trello
 - Freemium model

- Easy to get started
- Adhoc Teams
- Quick Trello Demo
- Mobile App
- Kanbanize
 - Comparable to JIRA
 - Multiple boards
 - Custom Fields
 - Templating, subtasks, analytics, events, time tracking, integrated messenger
 - Very Flexible, tags, flags, colors, custom fields etc
 - SaaS Model
 - Mobile App

Toyota's six rules:

- Later process picks up the number of items indicated by the kanban at the earlier process.
- Earlier process produces items in the quantity and sequence indicated by the kanban.
- No items are made or transported without a kanban.
- Always attach a kanban to the goods.
- Defective products are not sent on to the subsequent process. The result is 100% defect-free goods.
- Reducing the number of kanban increases the sensitivity.