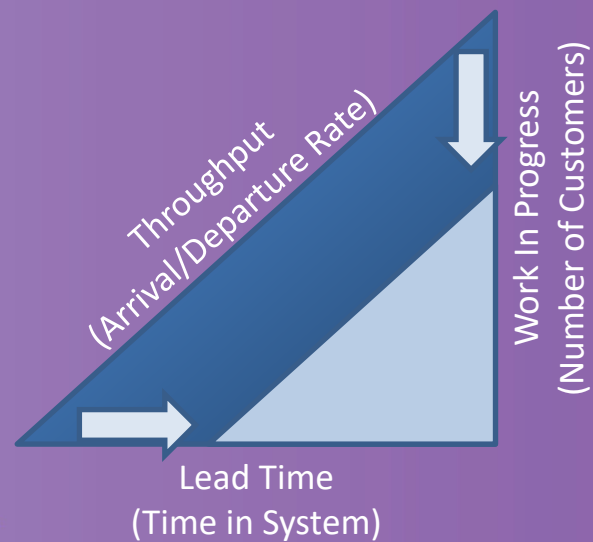
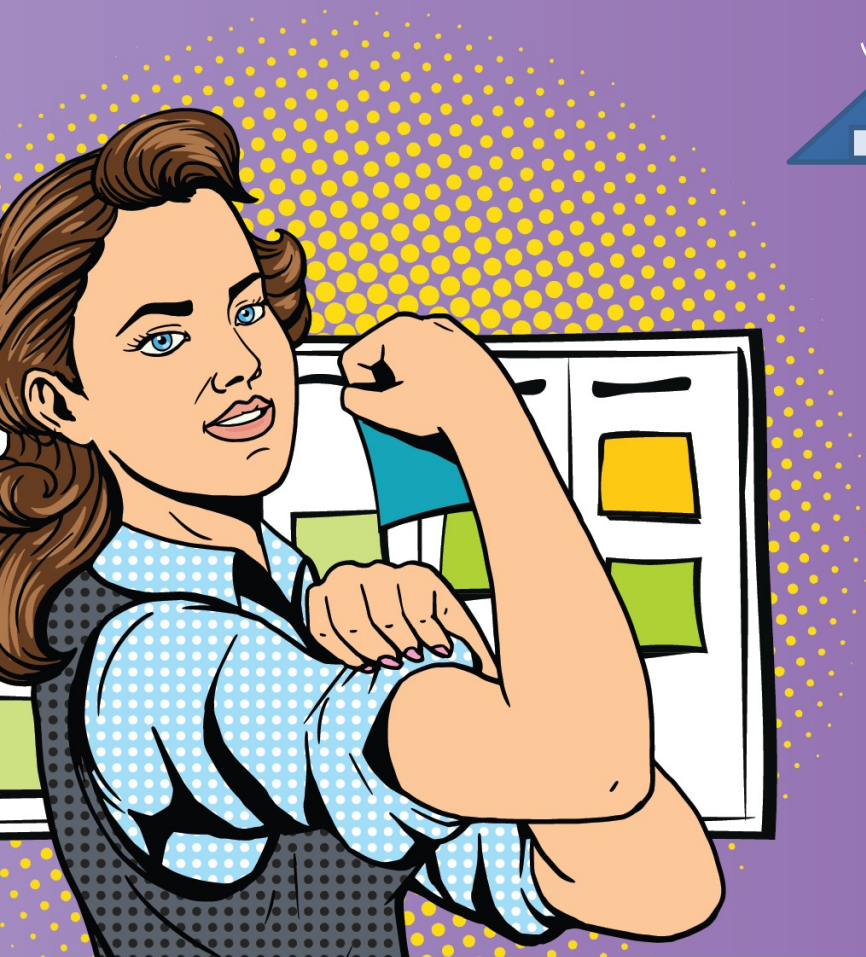




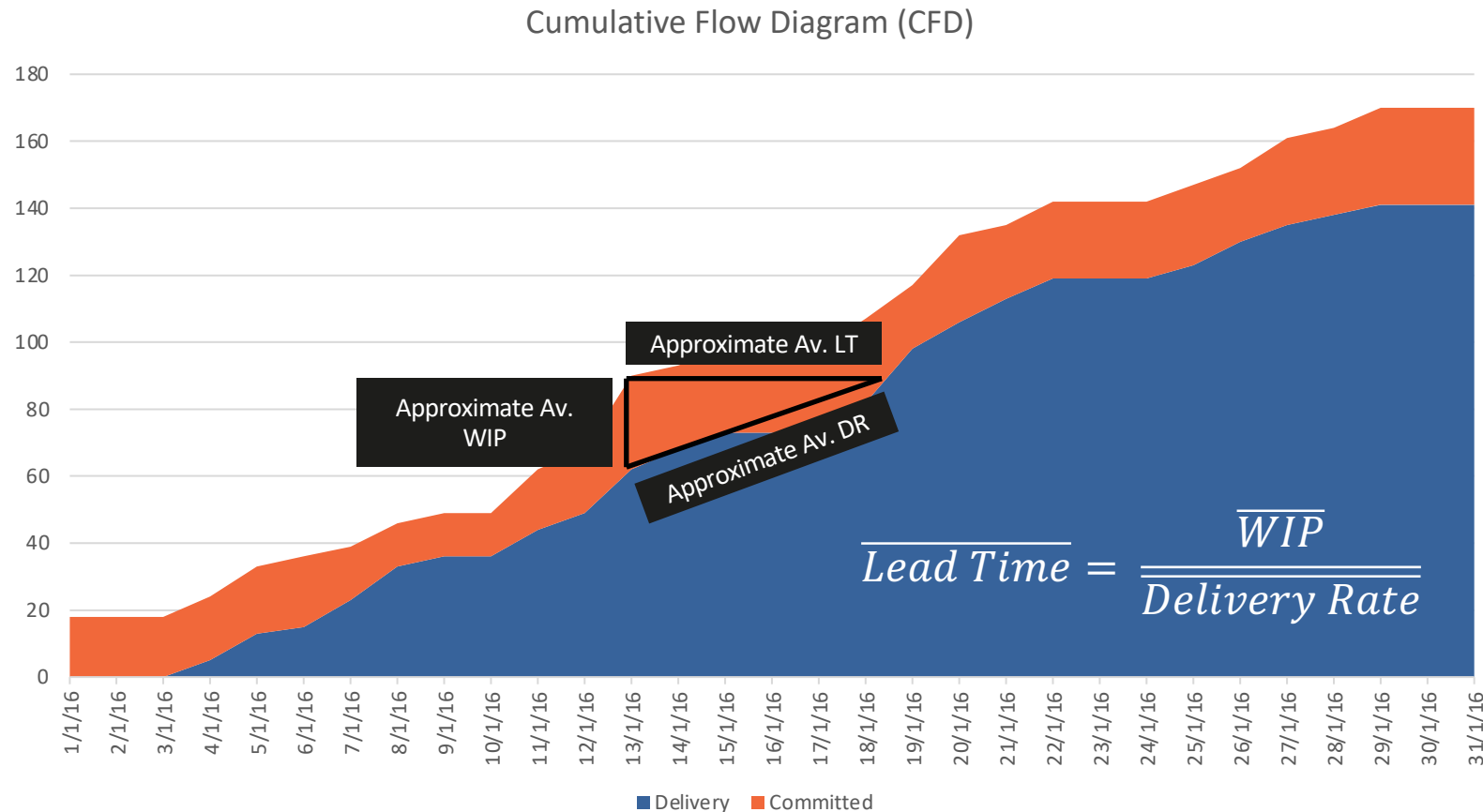
Exploring Little's Law

Todd Little
Chairman Kanban University

Estimate the # of Jelly Beans



Little's Law – a Relationship of Averages



Assumes that the system is not “trending” over the period of averaging.

The number of items entering and leaving system must be similar, and the average “age” of items in progress must not be increasing or decreasing.

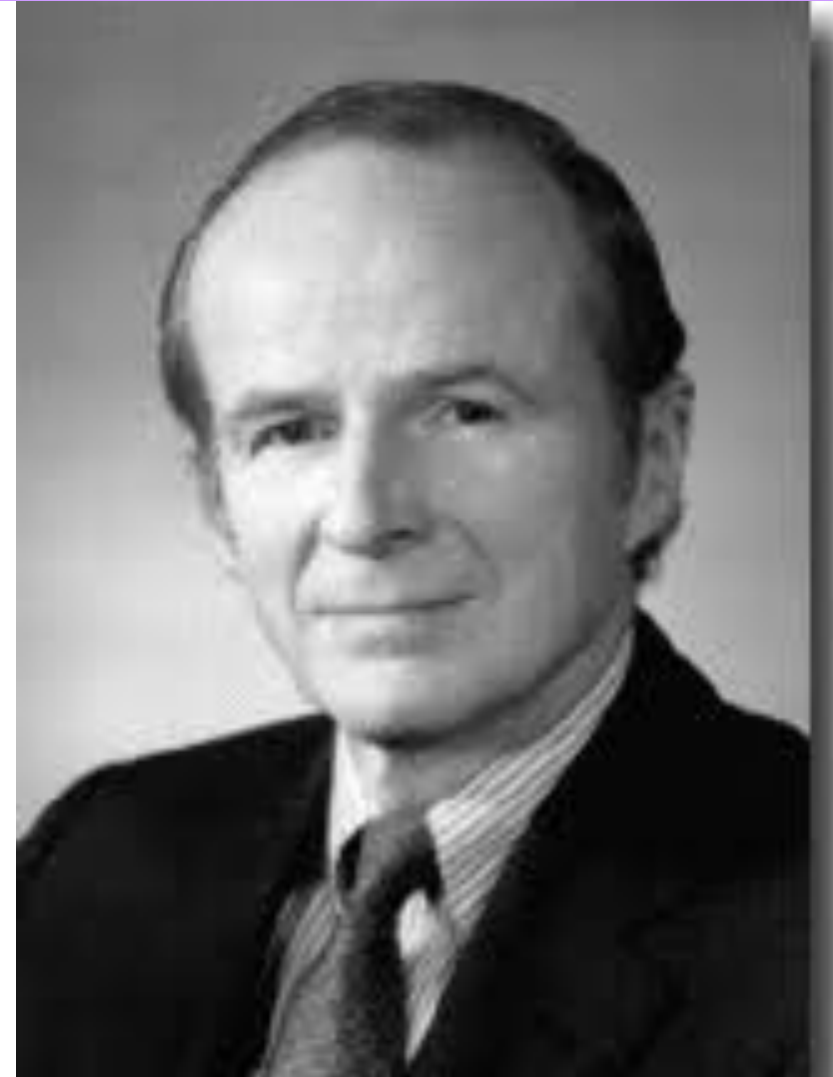
In spite of this restriction, the formula has proven its usefulness in many circumstances for understanding **how flow systems behave as WIP changes.**

Little's **L λ W** (John D. C. Little version)

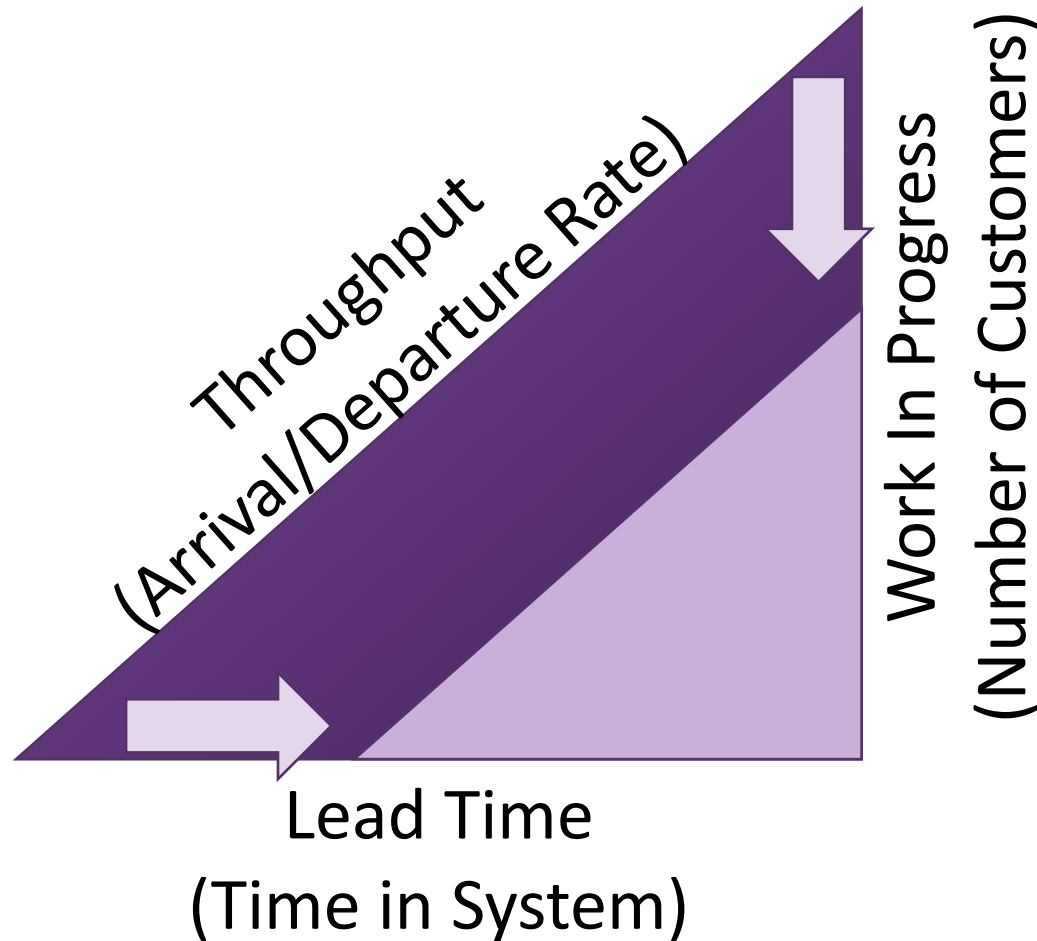
"The average number of customers in a system (over some interval) is equal to their average arrival rate, multiplied by their average time in the system."

$$\begin{array}{l} \text{Average} \\ \text{Length of} \\ \text{Queue} \end{array} \quad \begin{array}{l} \text{Average} \\ \text{Arrival Rate} \end{array} \quad \mathbf{L = \lambda W} \quad \begin{array}{l} \text{Average} \\ \text{Wait Time} \end{array}$$

$$\overline{\text{Delivery Rate}} = \frac{\overline{\text{WIP}}}{\overline{\text{Lead Time}}}$$



Little's Law in Knowledge Work



Little's Law Assumptions – The System is Stable and running at capacity

- The average Arrival Rate is equal to the average Departure Rate

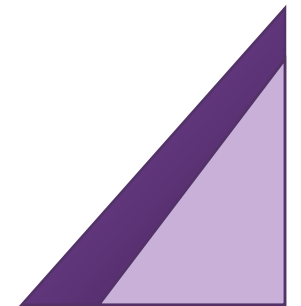
Cumulative Flow Diagrams



Stable system



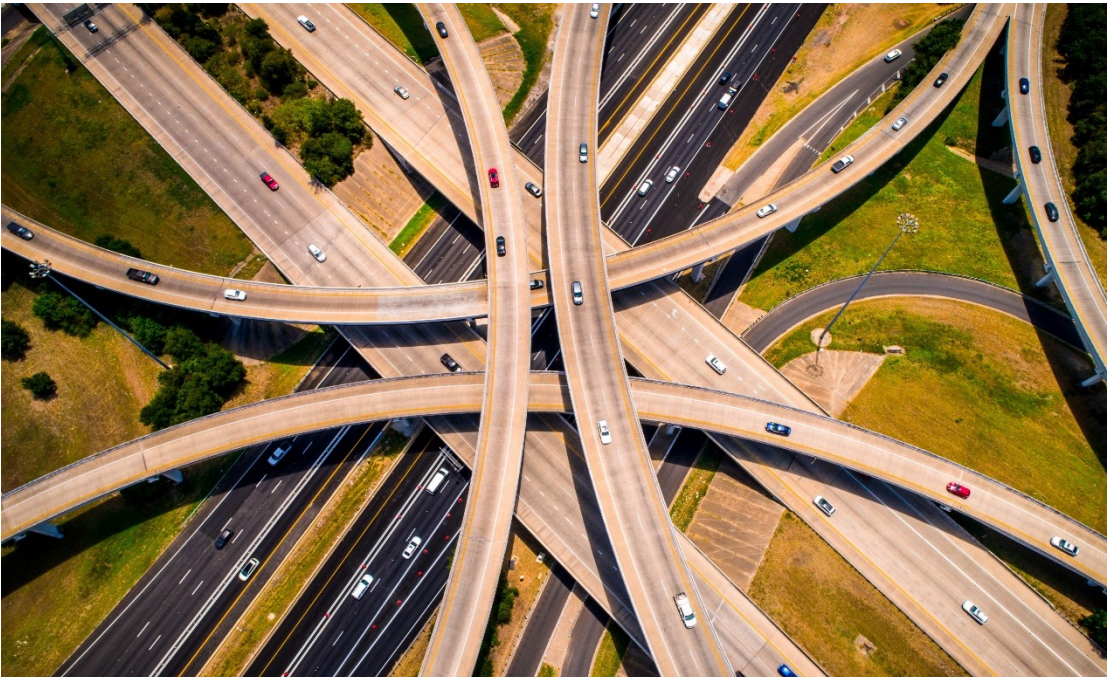
WIP increasing



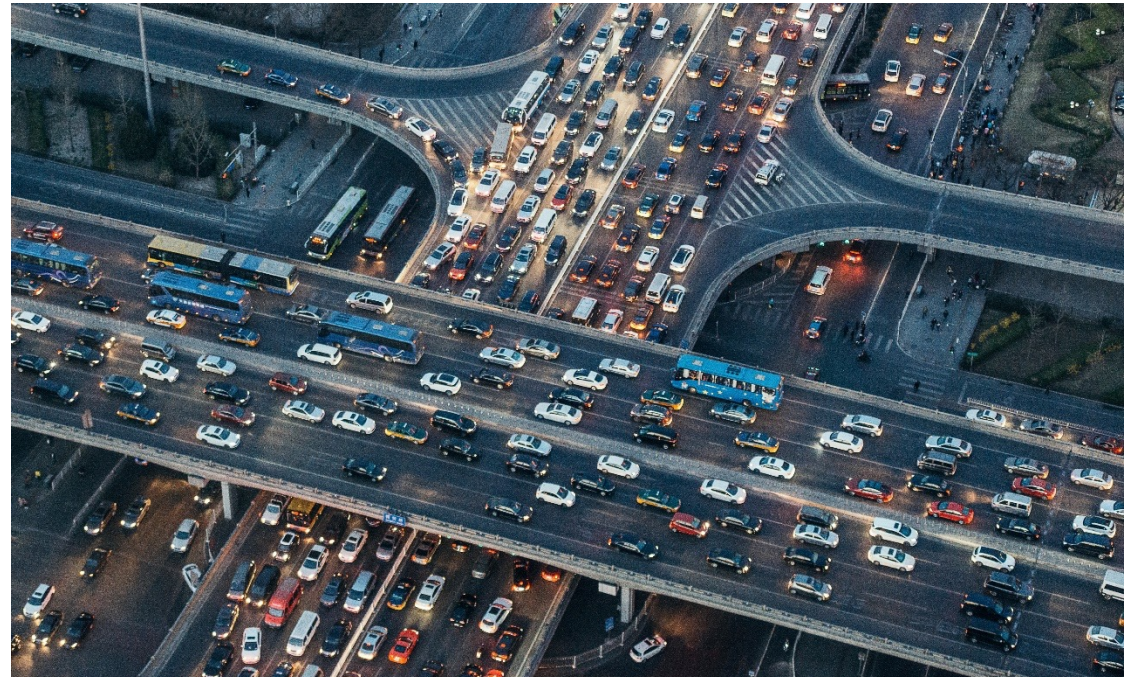
WIP decreasing

Introduction: Flow Efficiency

**Where would you prefer to travel?
How fast and smooth will your travel be?**



Few delays



Traveling time largely determined by
delays.

Little's Law in Action

**WIP: why limiting
work in progress
makes sense**

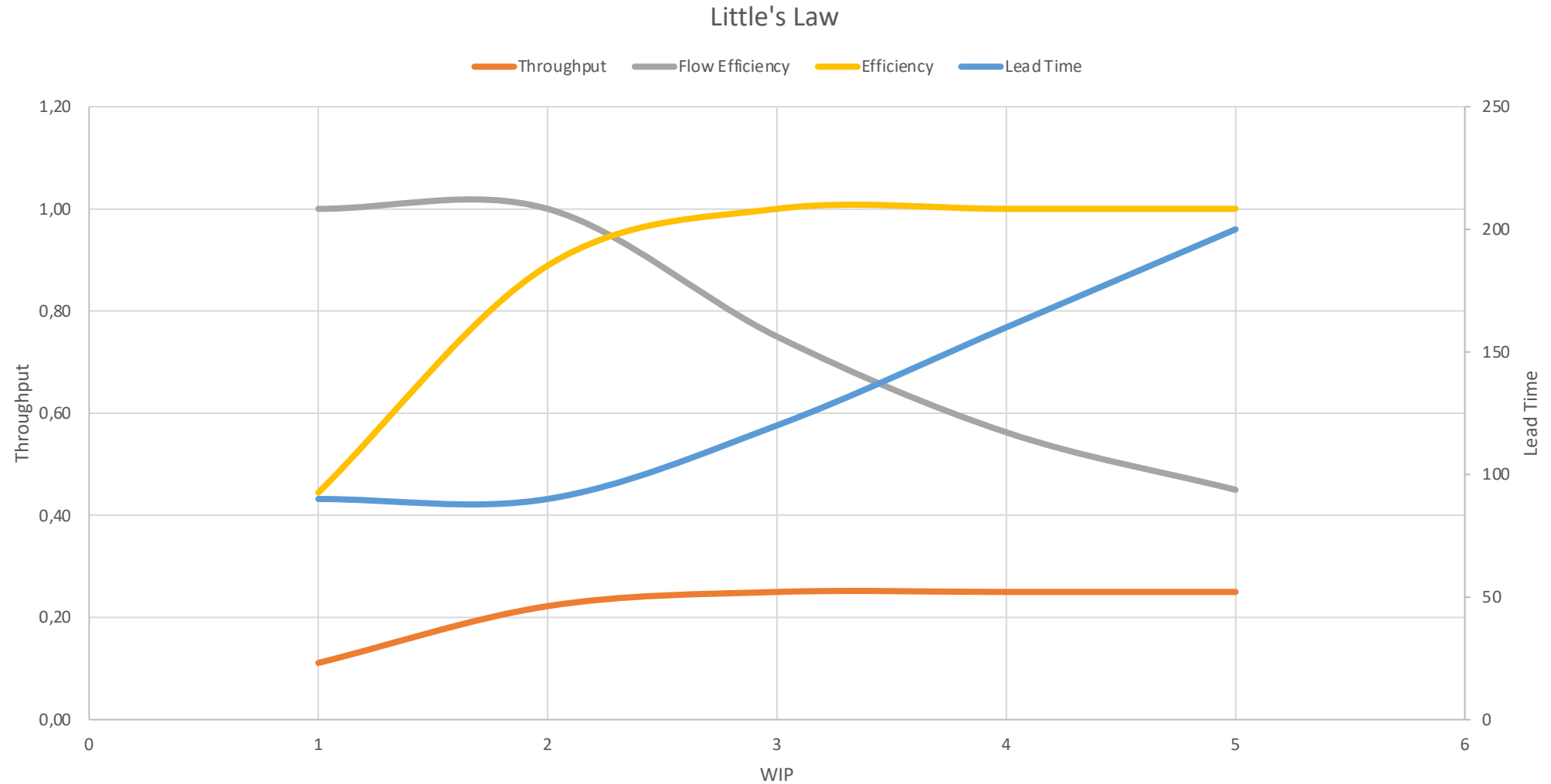
by David Lowe

ScrumAndKanban.co.uk

Summary of Video Results

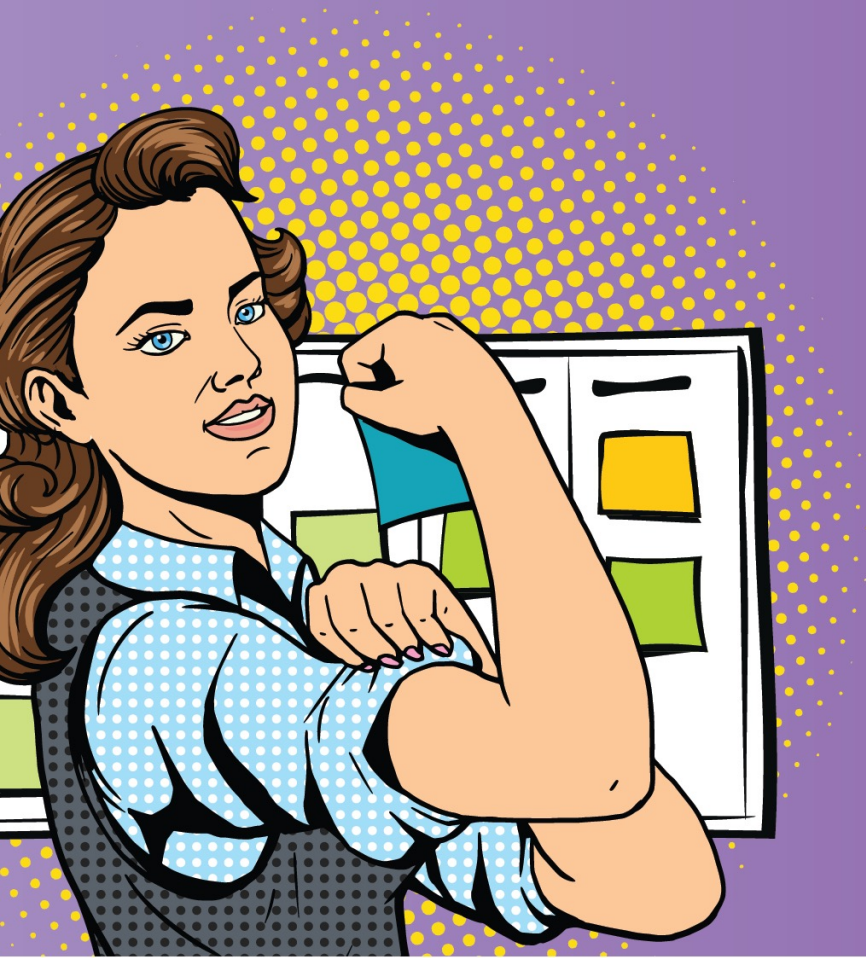
WIP	Lead Time	Throughput	Flow Efficiency	Efficiency
1	90	0.11	1.00	0.44
2	90	0.22	1.00	0.89
3	120	0.25	0.75	1.00
4	160	0.25	0.56	1.00
5	200	0.25	0.45	1.00

Flow Efficiency, Efficiency, Lead Time and Throughput





What does Little's Law say about multitasking?



Design for Flow

Be Like Water - Continuity Equation



$$Q = 10 \text{ m}^3/\text{sec} \quad A = 5 \text{ m}^2 \quad v = 2 \text{ m/sec}$$

$$Q = A * v$$

5x1

5x2

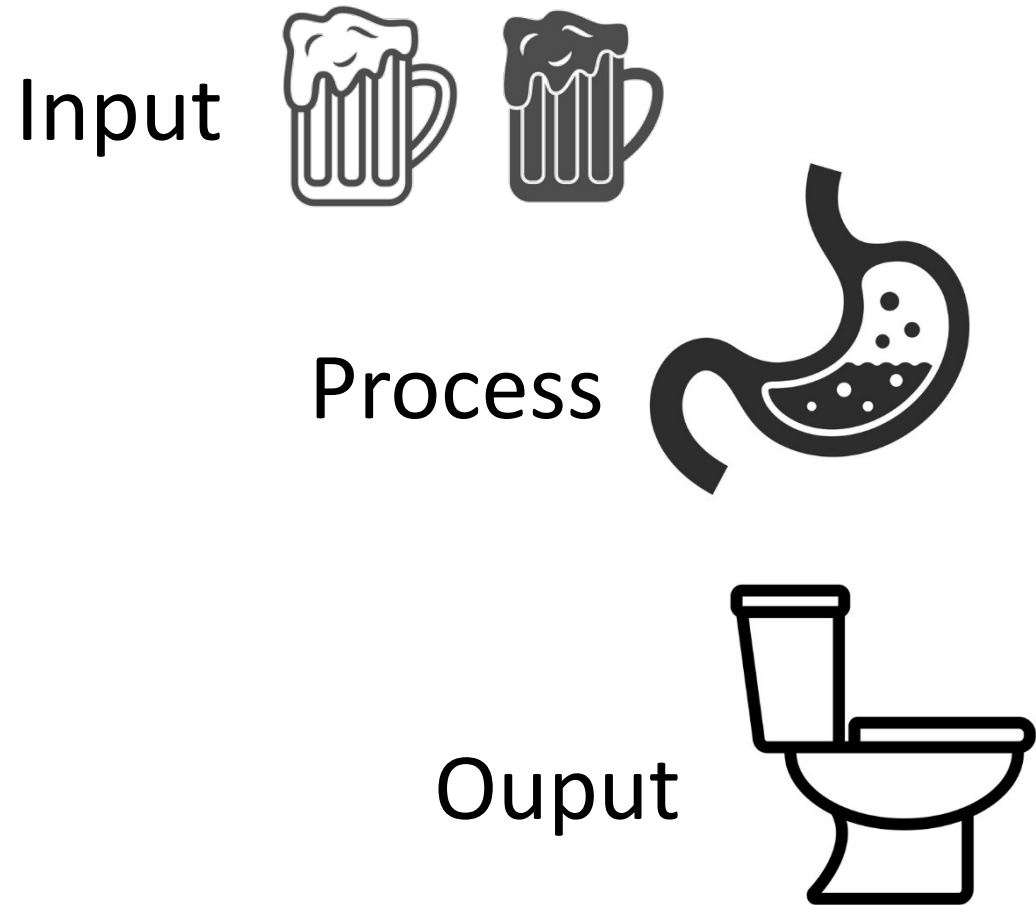
$$Q = 10 \text{ m}^3/\text{sec} \quad A = 10 \text{ m}^2 \\ v = 1 \text{ m/sec}$$

10x1

$$Q = 10 \text{ m}^3/\text{sec} \quad A = 10 \text{ m}^2 \quad v = 1 \text{ m/sec}$$

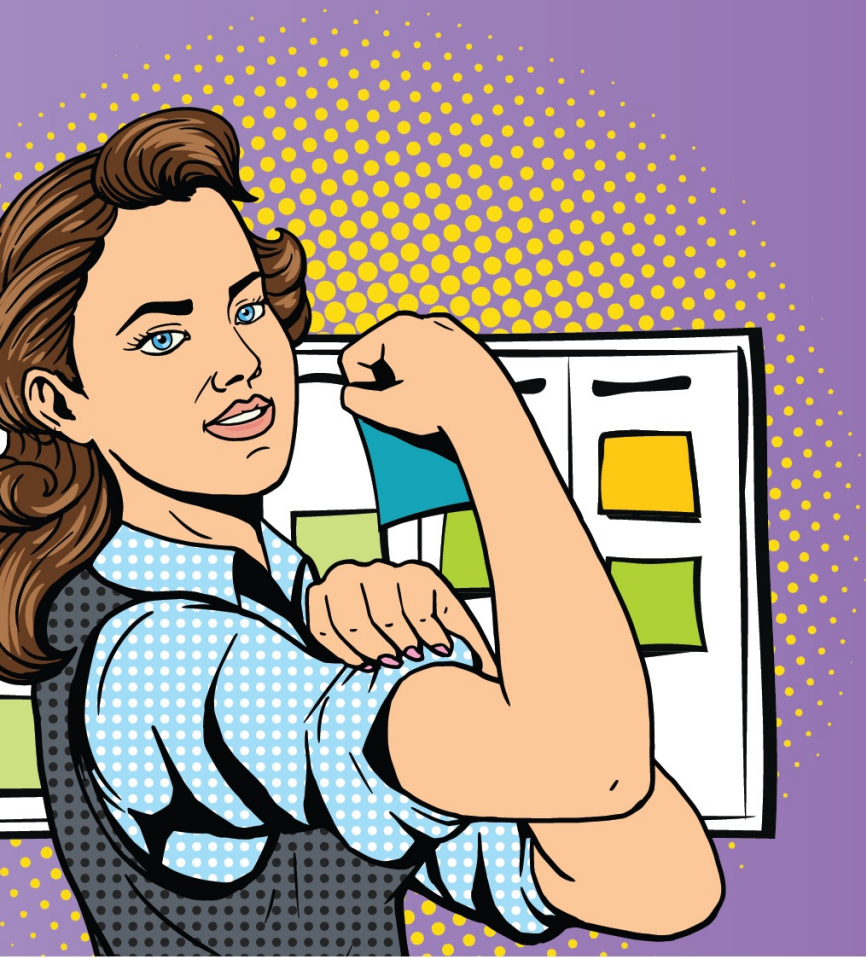
Be Like Water – College Chem E Version

Input – Output = Accumulation



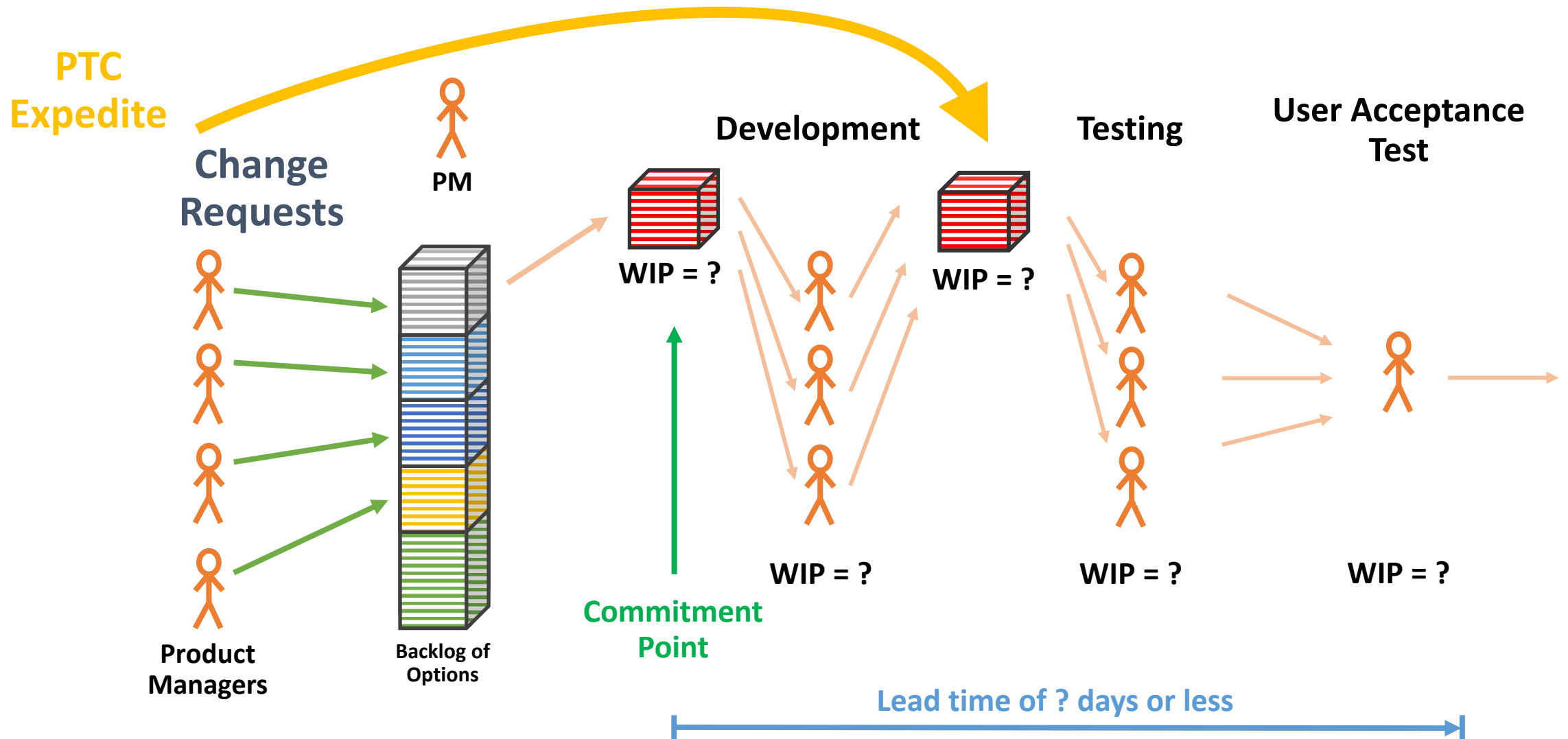
Accumulation



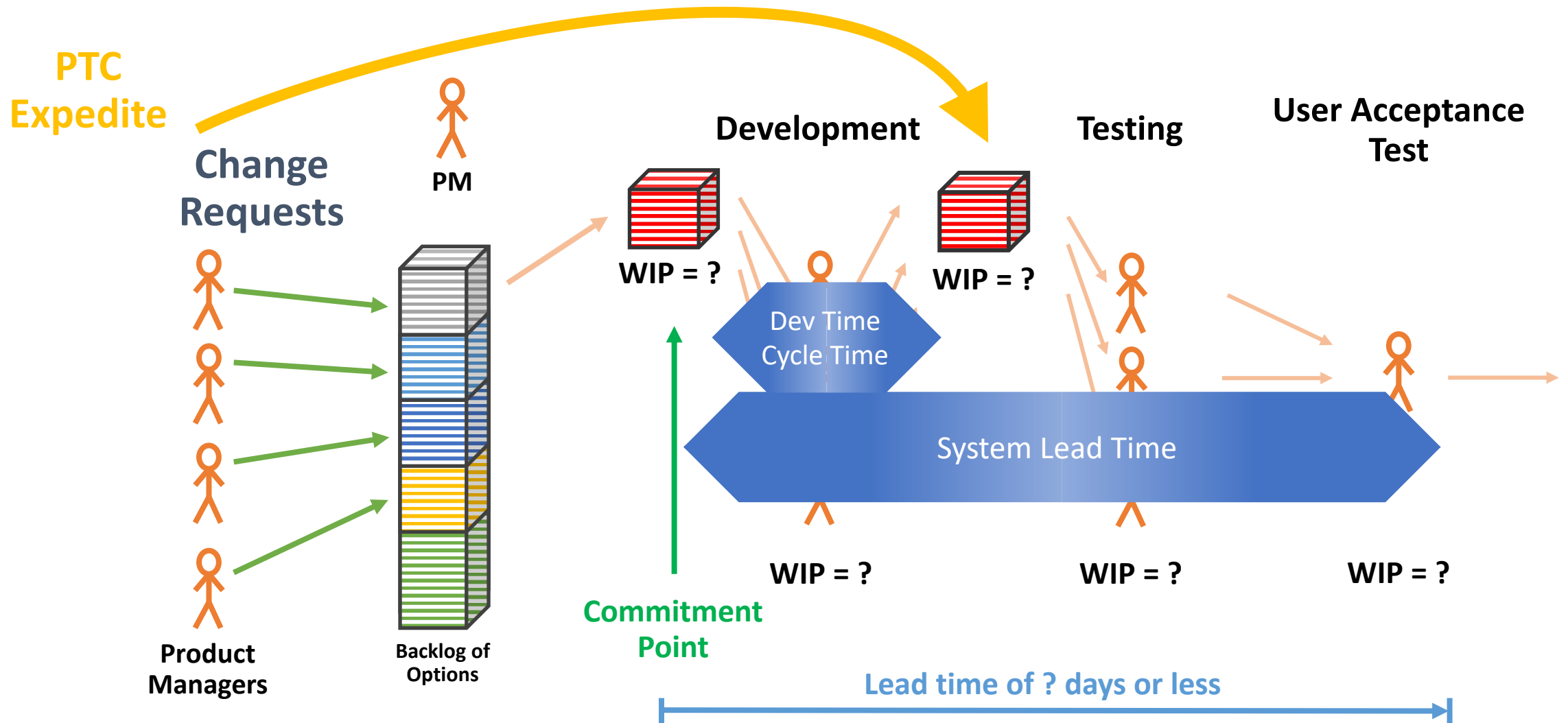


Smooth Flow Throughout Optimize the Bottleneck

Kanban System Design for XIT

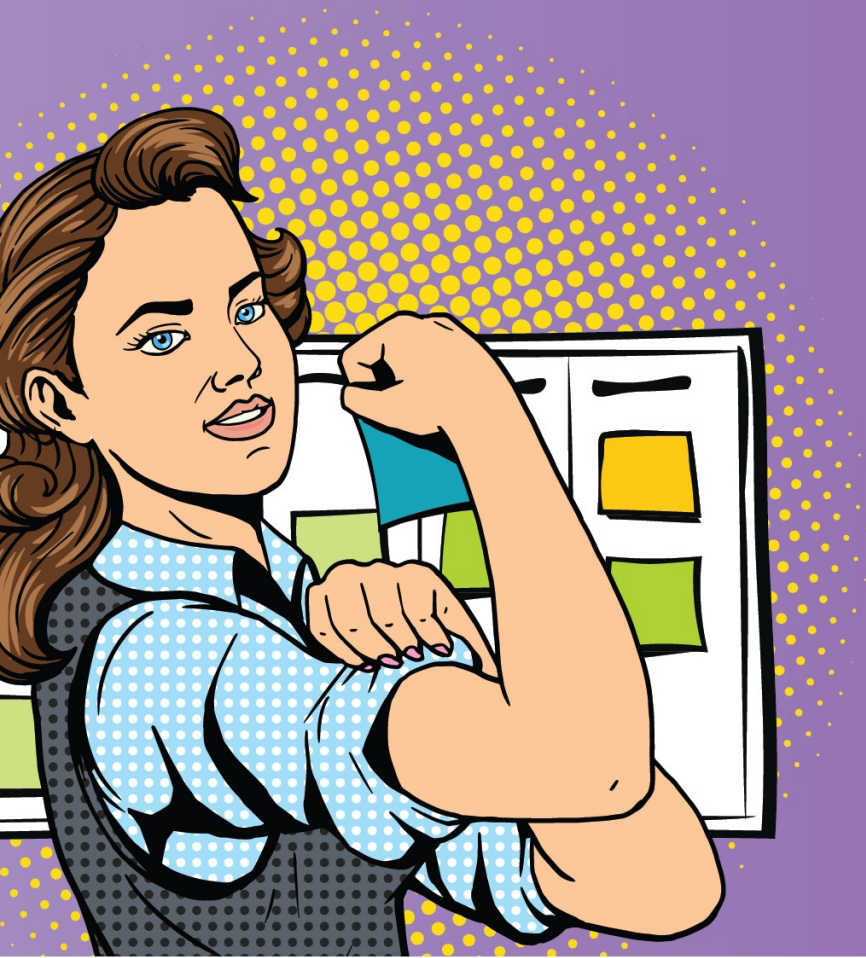


Kanban System Design for XIT



Design Rules

- Start with what data you have. It need not be perfect
- Design for the bottleneck
 - For XIT - Engineering and Test
- Throughput is constant through the flow
- Queues – large enough to prevent starvation, small enough to keep lead time small
- Difference between WIP Limit and Average WIP



Let's get started

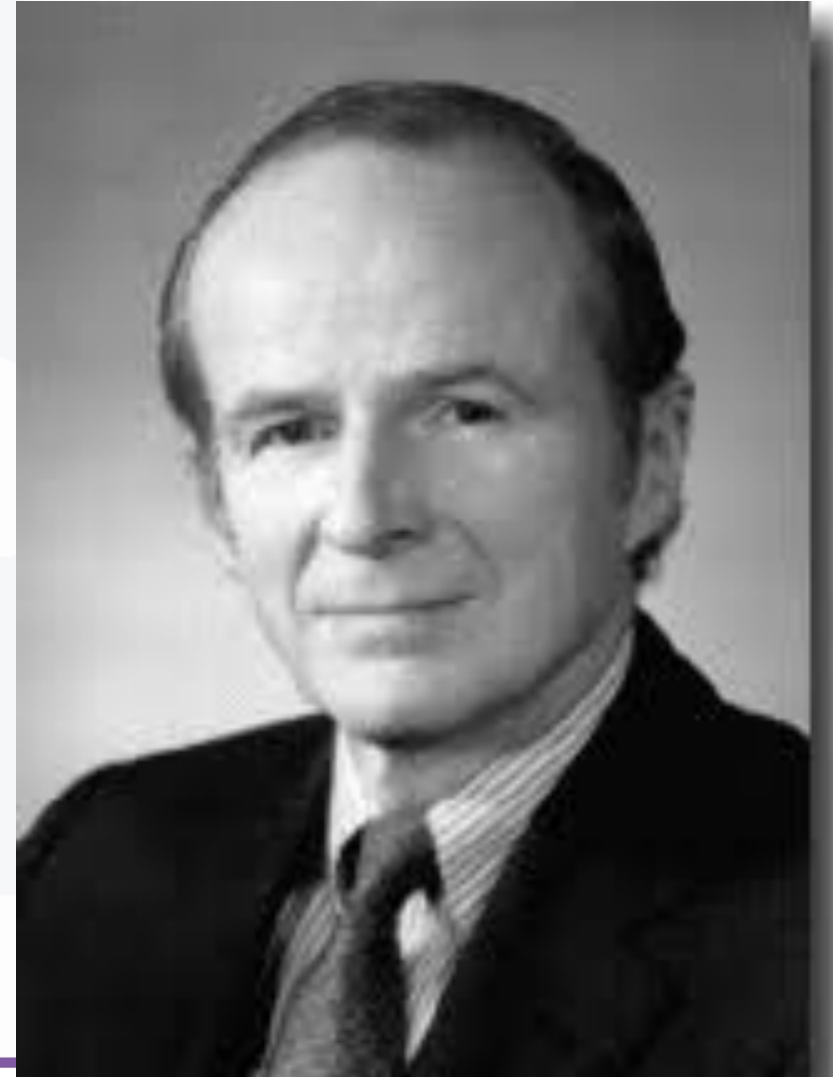
Little's **LλW** (John D. C. Little version)

$$\overline{\text{Delivery Rate}} = \frac{\overline{\text{WIP}}}{\text{Lead Time}}$$

Not Little's Law, but sometimes useful

$$\overline{\text{Delivery Rate}} = \frac{\text{Max WIP}}{\text{Max Lead Time}}$$

$$\text{Min Delivery Rate} = \frac{\overline{\text{WIP}}}{\text{Max Lead Time}}$$

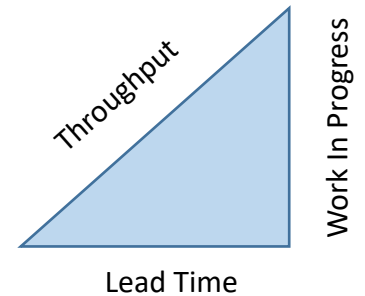


Little's Law @XIT

	System		Ready	Dev	Dev Done	Test	UAT
Ave Throughput							
WIP Average							
Ave Lead Time							
WIP Limit							
Max Lead Time							

Determining an Expectation of Service Delivery

- Engineering effort (touch time)
 - Average engineering time was 11 days with outliers > 24 days
- Demand rate is 10-12 CR per month – can we meet it?
- Delivery rate estimate
 - Without estimation interruptions or multitasking, we have:
 - WIP = 6 (3 dev + 3 test)
 - Average Lead Time = 11 days
 - $Average\ Throughput = \frac{Average\ WIP}{Average\ LT} = \frac{6}{11} = 0.55\ items/day = 11\ items/mo$
- Queuing & buffering time estimates
 - Queues: large enough to prevent starvation, small enough to keep Lead Time small.



Little's Law @XIT

	System		Ready	Dev	Dev Done	Test	UAT
Ave Throughput				0.55		0.55	
WIP Average				3		3	
Ave Lead Time				5.5		5.5	
WIP Limit				3		3	
Max Lead Time				12		12	

Little's Law @XIT - Smooth flow throughout

	System		Ready	Dev	Dev Done	Test	UAT
Ave Throughput	0.55		0.55	0.55	0.55	0.55	0.55
WIP Average				3		3	
Ave Lead Time				5.5		5.5	
WIP Limit				3		3	
Max Lead Time				12		12	

Queues and WIP limits

- Ready Queue
 - Replenishment will be weekly.
 - Allow 2X buffer
 - What is Starting WIP, Ending WIP, Average WIP?
 - Average Throughput of 2.7 items/week (5×0.55)
 - Start with WIP limit of 5 to allow 2x buffer
 - Beginning WIP of 5, Ending WIP of $5 - 2.7 = 2.3$
 - Average WIP of $5 - (5 - 2.7) / 2 = 3.65$

Little's Law @XIT - Replenishment

	System		Ready	Dev	Dev Done	Test	UAT
Ave Throughput	0.55		0.55	0.55	0.55	0.55	0.55
WIP Average			3.65	3		3	
Ave Lead Time			6.6	5.5		5.5	
WIP Limit			5	3		3	
Max Lead Time			10	12		12	

Queues and WIP limits

- Dev Done Queue
 - 3 items will be in Test with average Lead Time of 6 days.
 - How frequent does a Test slot free up?
 - On average, Test will free one slot every 2 days
 - Assume average Lead Time of 2 in Dev Done Queue

Little's Law @XIT - Dev Done Buffer

	System		Ready	Dev	Dev Done	Test	UAT
Ave Throughput	0.55		0.55	0.55	0.55	0.55	0.55
WIP Average			3.65	3	1.1	3	
Ave Lead Time			6.6	5.5	2	5.5	
WIP Limit			5	3	2	3	
Max Lead Time			10	12	3.6	12	

Queues and WIP limits

- UAT
 - Max Lead Time of 10 days.
 - WIP Limit of $10 * 0.55 = 5.5 \rightarrow 6$
 - Assume Average WIP of $\frac{1}{2}$ limit = 3

Little's Law @XIT - UAT Buffer

	System		Ready	Dev	Dev Done	Test	UAT
Ave Throughput	0.55		0.55	0.55	0.55	0.55	0.55
WIP Average			3.65	3	1.1	3	3
Ave Lead Time			6.6	5.5	2	5.5	5.5
WIP Limit			5	3	2	3	6
Max Lead Time			10	12	3.6	12	10

Little's Law @XIT - System

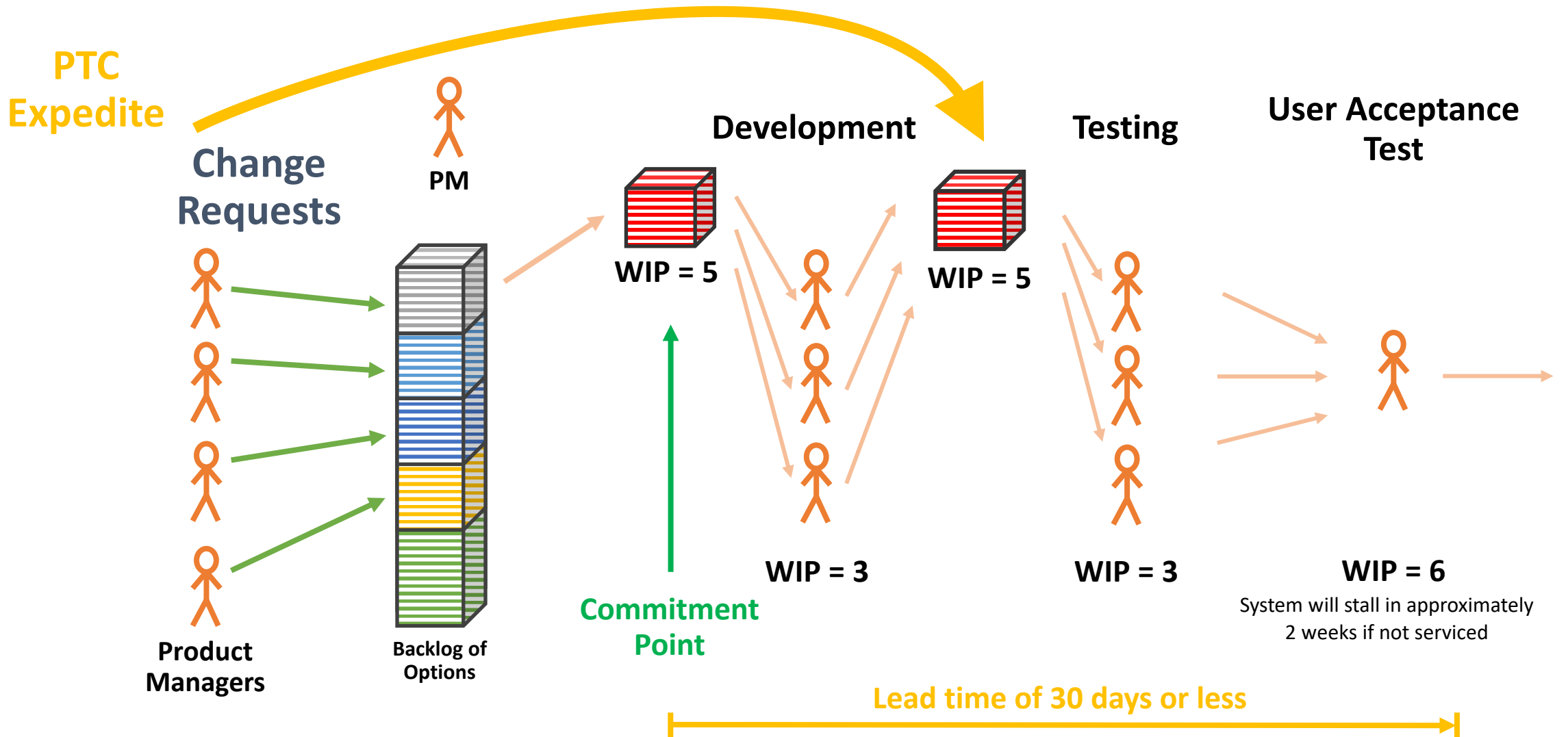
	System		Ready	Dev	Dev Done	Test	UAT
Ave Throughput	0.55		0.55	0.55	0.55	0.55	0.55
WIP Average	13.7		3.65	3	1.1	3	3
Ave Lead Time	25.1		6.6	5.5	2	5.5	5.5
WIP Limit	19		5	3	2	3	6
Max Lead Time	47.6		10	12	3.6	12	10

Can we meet a target Lead Time of 30 days with a high (98%) Service Expectation?

What knobs might you tweak to help meet the SLE?

	System		Ready	Dev	Dev Done	Test	UAT
Ave Throughput	0.55		0.55	0.55	0.55	0.55	0.55
WIP Average	13.7		3.65	3	1.1	3	3
Ave Lead Time	25.1		6.6	5.5	2	5.5	5.5
WIP Limit	19		5	3	2	3	6
Max Lead Time	47.6		10	12	3.6	12	10

Kanban System Design for XIT



Let's play a game



Google Spreadsheet

bit.ly/3PGymF9

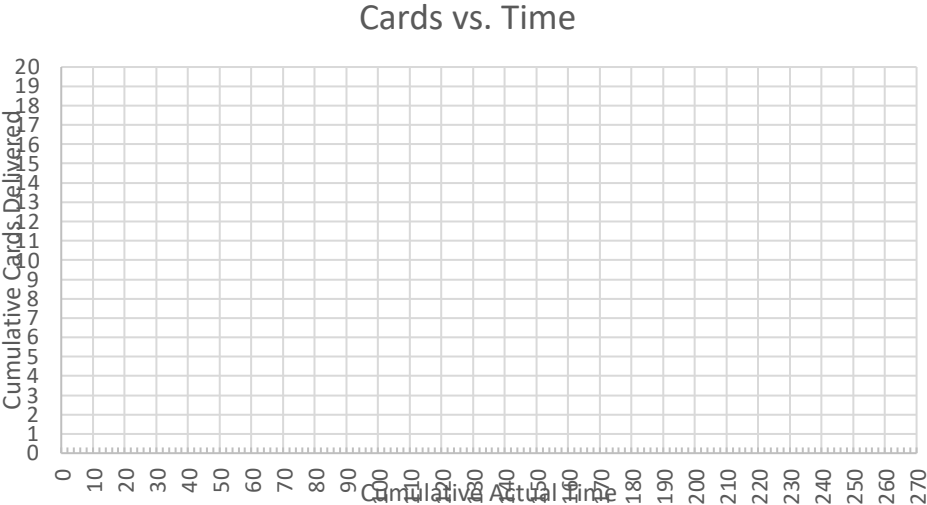
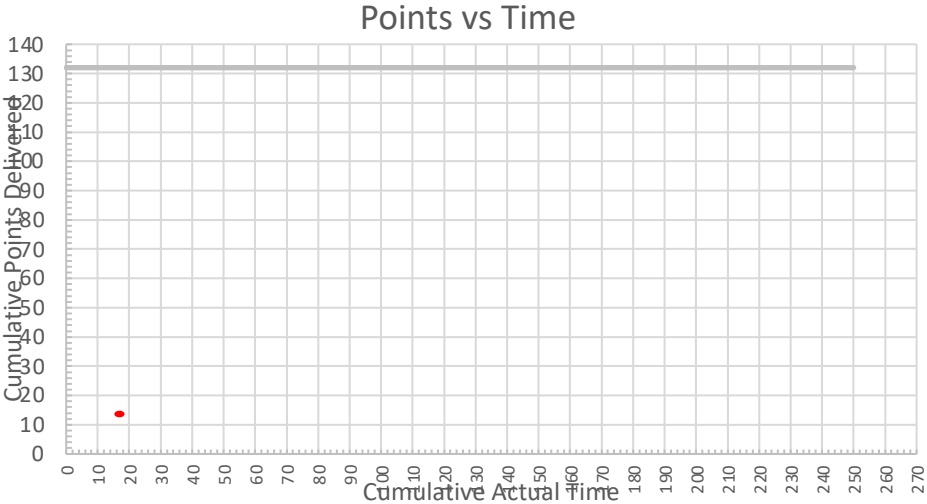


Old School

Little's Law Tally Sheet							
Y(cards)				X	Y(points)	=Cum Actual (E)*132 / Cum Size (F)	=Cum Actual (E) *20 / # Cards (A)
Card	Size	Die roll	Actual Time	Cummulative Actual	Cummulative Size	Size Forecast	Count Forecast
1							
2							
3							
4							
5							
6							
7							
8							
9							
10							
11							
12							
13							
14							
15							
16							
17							
18							
19							
20							

Serious Old School

Little's Law Tally Sheet							
Y(cards)				X	Y(points)	=Cum Actual (E)*132 / Cum Size (F)	=Cum Actual (E) *20 / # Cards (A)
Card	Size	Die roll	Actual Time	Cummulative Actual	Cummulative Size	Size Forecast	Count Forecast
1							
2							
3							
4							
5							
6							
7							
8							
9							
10							
11							
12							
13							
14							
15							
16							
17							
18							
19							
20							



Per Table

Deck of Cards

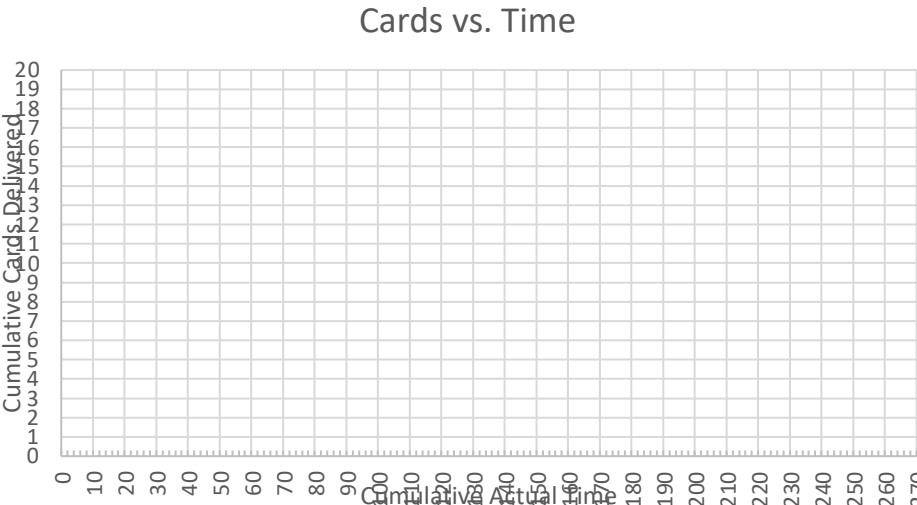
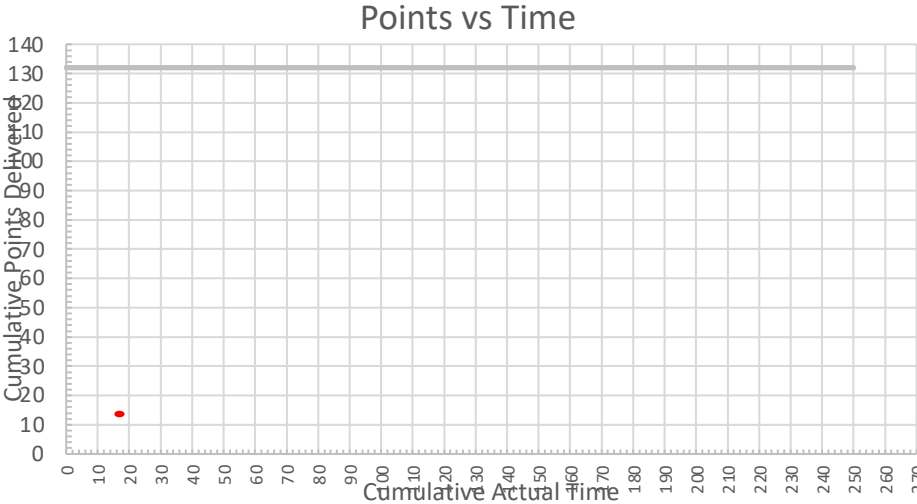


Adjustment Table

Card Value						
Die	2	3	5	8	13	20
1	1	2	3	4	8	13
2	1	2	4	6	10	16
3	2	3	5	8	13	20
4	2	3	5	8	13	20
5	4	6	7	11	16	25
6	6	8	10	16	22	30

Per person

Little's Law Tally Sheet							
Y(cards)				X	Y(points)	=Cum Actual (E)*132 / Cum Size (F)	=Cum Actual (E) *20 / # Cards (A)
Card	Size	Die roll	Actual Time	Cummulative Actual	Cummulative Size	Size Forecast	Count Forecast
1							
2							
3							
4							
5							
6							
7							
8							
9							
10							
11							
12							
13							
14							
15							
16							
17							
18							
19							
20							



Background

Each card represents a story in the backlog.

They have been estimated through rigorous Planning Poker

We will process cards sequentially,

but each table will have a different order

Please shuffle the deck of cards (but not after each draw)

Cards will then be pulled sequentially

Each person will implement the story by rolling their die

The die will tell us how accurate the estimate was and how long it really takes

Example:

Draw a card to be used by everyone at the table

13

Now each person rolls their die



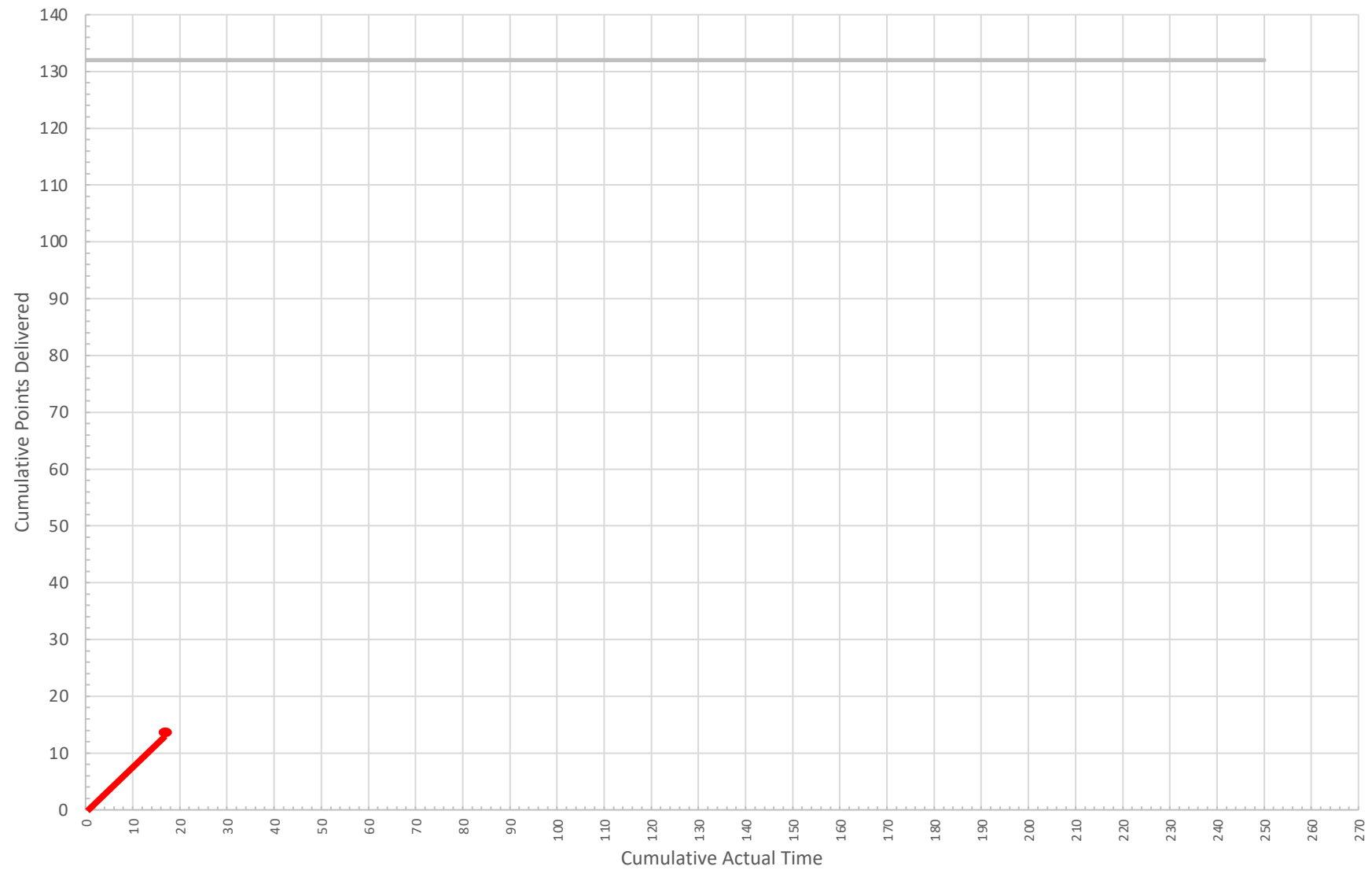
Adjustment Table

Card Value						
Die	2	3	5	8	13	20
1	1	2	3	4	8	13
2	1	2	4	6	10	16
3	2	3	5	8	13	20
4	2	3	5	8	13	20
5	4	6	7	11	16	25
6	6	8	10	16	22	30

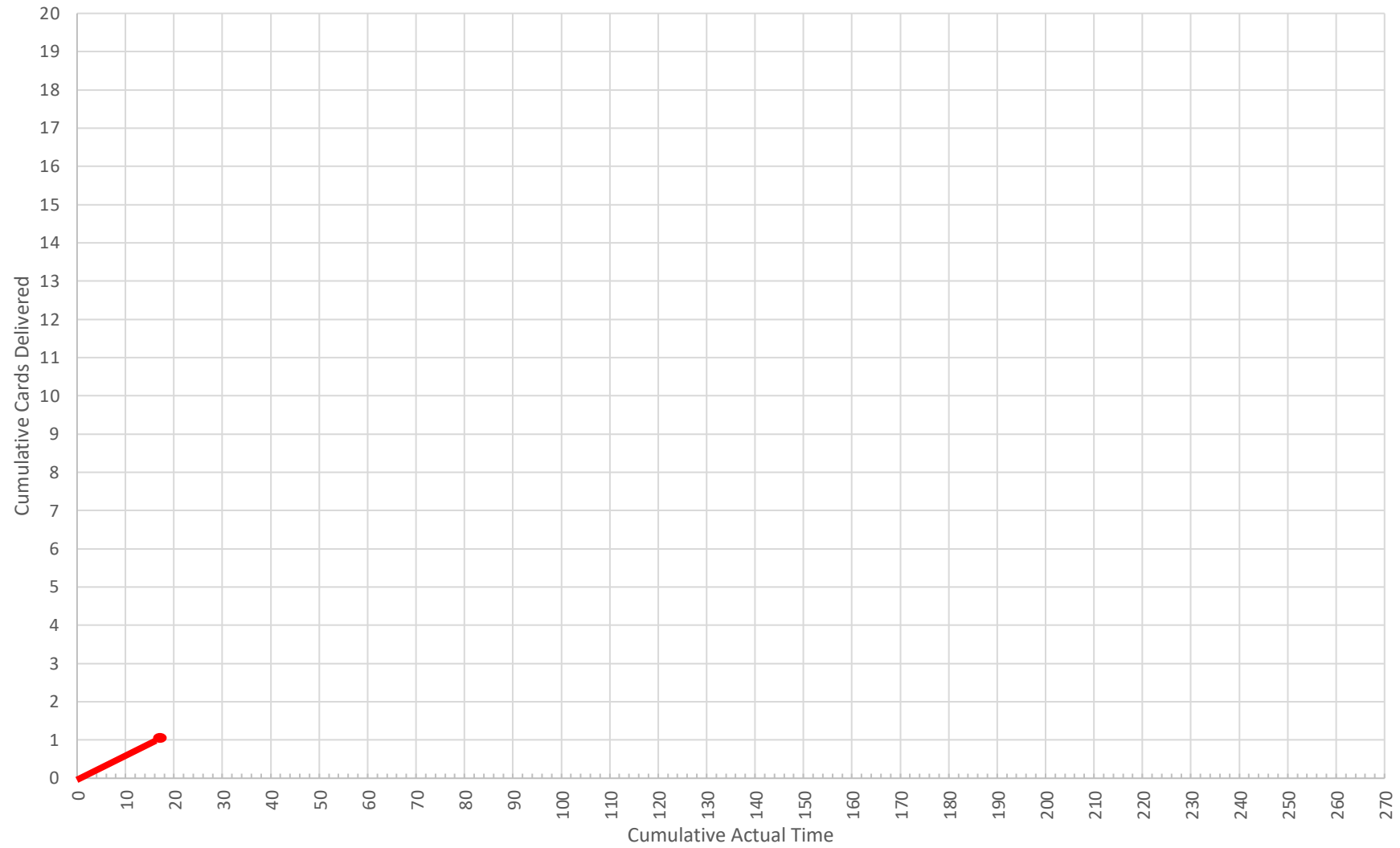
Little's Law Tally Sheet

Y(cards)				X	Y(points)	=Cum Actual (E)*132 / Cum Size (F)	=Cum Actual (E) *20 / # Cards (A)
Card	Size	Die roll	Actual Time	Cumulative Actual	Cumulative Size	Size Forecast	Count Forecast
1	13	5	16	16	13		
2							
3	Card		Lookup		Sum Size		
4		Die		Sum Actual			
5							
6							
7							
8							
9							
10							

Points vs Time



Cards vs. Time



Google Spreadsheet

bit.ly/3PGymF9

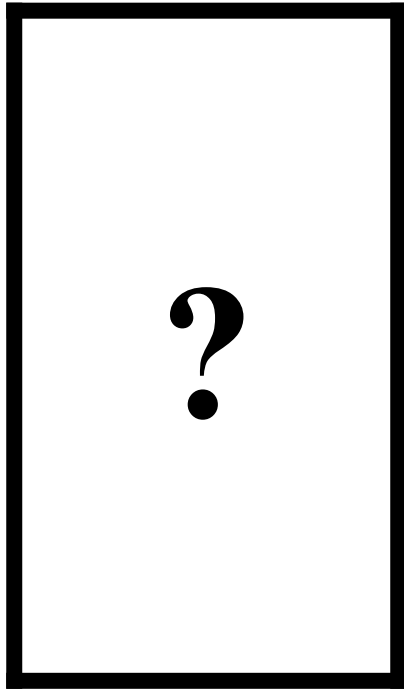


Old School

Little's Law Tally Sheet							
Y(cards)				X	Y(points)	=Cum Actual (E)*132 / Cum Size (F)	=Cum Actual (E) *20 / # Cards (A)
Card	Size	Die roll	Actual Time	Cummulative Actual	Cummulative Size	Size Forecast	Count Forecast
1							
2							
3							
4							
5							
6							
7							
8							
9							
10							
11							
12							
13							
14							
15							
16							
17							
18							
19							
20							

Let's do 1 card

Draw a card to be used by everyone at the table



Now each person rolls their die

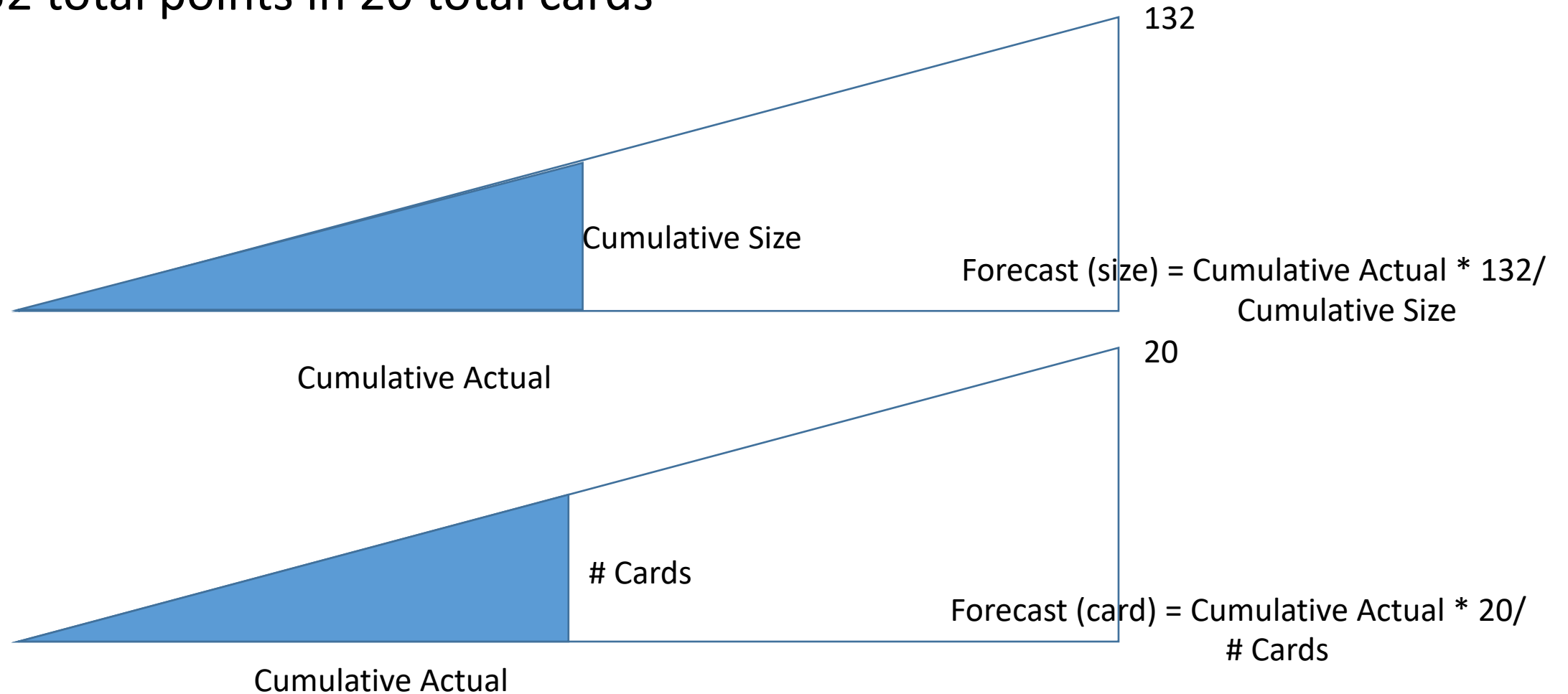


Task switching – Sometimes batches are better

- Run through 5 cards recording on the Tally Sheet
- Calculate the Forecast based on the trends of the first 5 cards
 - Size Forecast = $\text{Cum Actual (E)} * 132 / \text{Cum Size (F)}$
 - Count Forecast = $\text{Cum Actual (E)} * 20 / \# \text{ Cards (A)}$
- Continue with the remaining 15 cards in batches of 5
- Old School – plot away

Forecasting

132 total points in 20 total cards



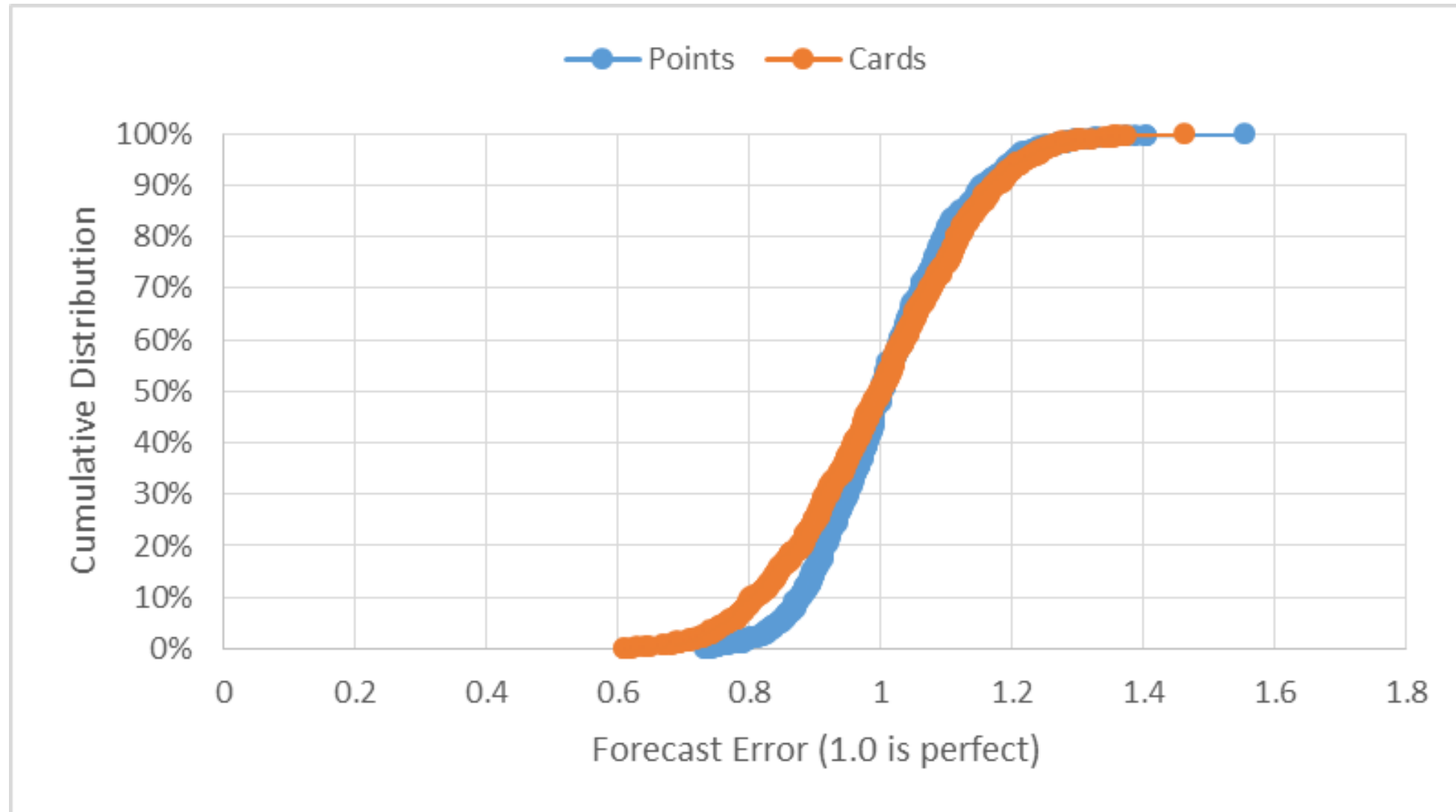
Little's Law Tally Sheet

Y(cards)				X	Y(points)	=Cum Actual (E)*132 / Cum Size (F)	=Cum Actual (E) *20 / # Cards (A)
Card	Size	Die roll	Actual Time	Cumulative Actual	Cumulative Size	Size Forecast	Count Forecast
1	13	5	16	16	13		
2	5	3	5	21	18		
3	8	2	6	27	26		
4	2	6	6	33	28		
5	5	1	3	36	33	144	144
6							
7							
8							
9							
10							

Final Summary

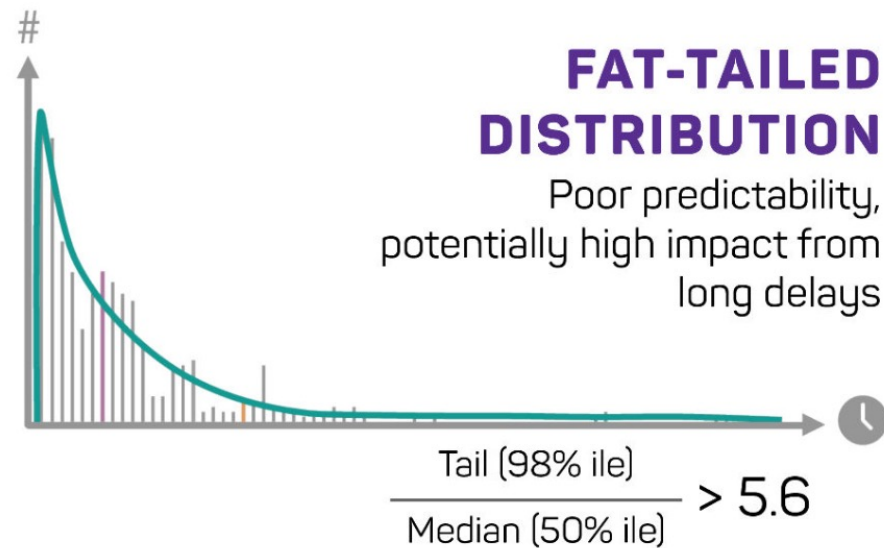
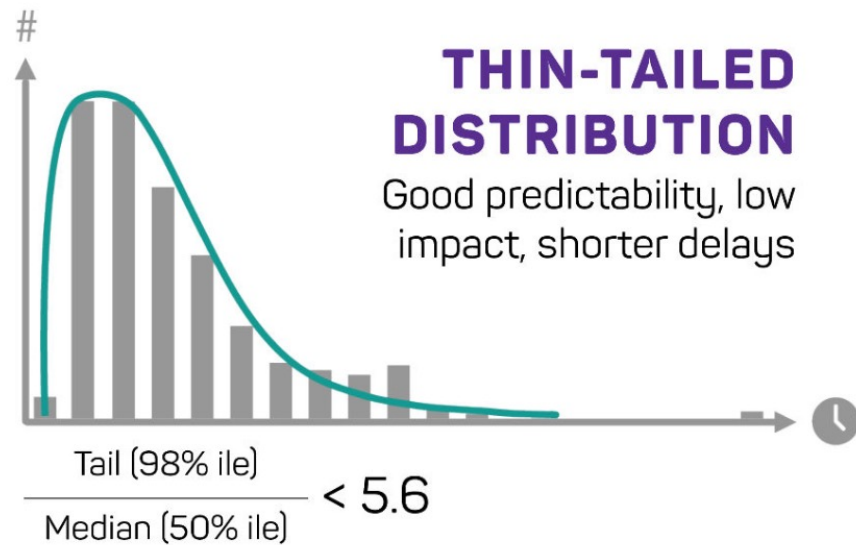
Card	Size	Die roll	Actual Time	Cumulative Actual	Cumulative Size	Size Forecast	Count Forecast
5							
10							
15							
20							

Monte Carlo Simulation Results



Data needs to be Thin-tailed for Little's Law

NATURE OF LEAD TIME



$$P90/P10 < 20$$

Estimate the # of Jelly Beans

