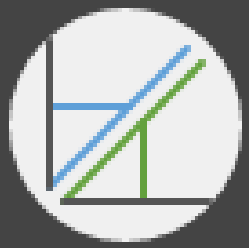


ActionableAgile
Take Control

Why Size Doesn't Matter

Daniel S. Vacanti
ActionableAgile™
daniel@actionableagile.com
@danvacanti

“When will it be done?”

“When will it be done?”

Date (number of days)

Elapsed Time

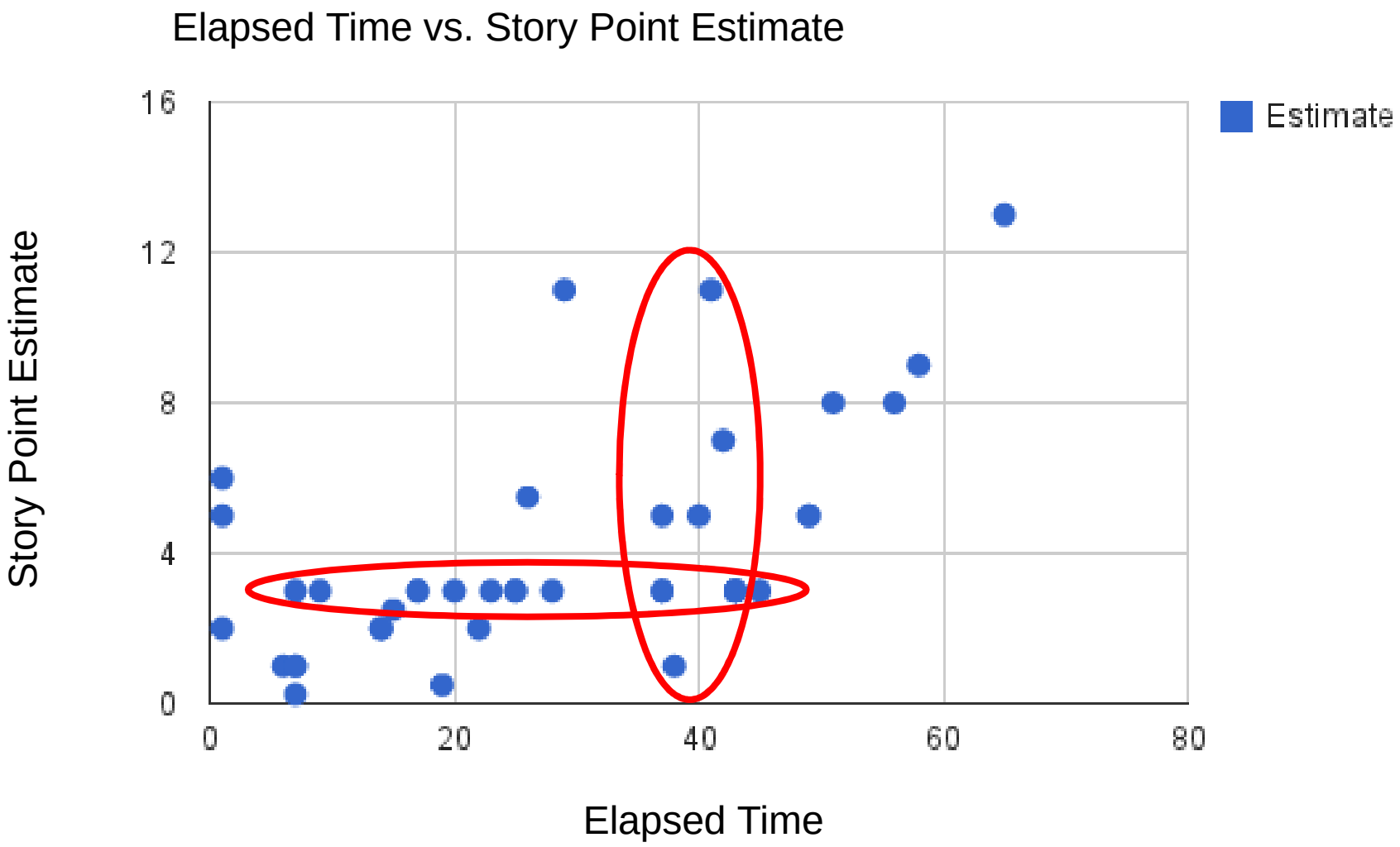
“When will it be done?”

Date (number of days)

Story Points / Velocity?

Stop me if you've heard this one before…

“Relative Complexity is the best predictor of how long it takes an item to complete”


Elapsed Time vs. Story Point Estimate
Story Point Estimate
16
12
8
4
0
0
20
40
60
80
Elapsed Time
Estimate

Year 2014

	Stories		Points	Average Days					
				InProgress	Ready for QA	QA	Ready for Aceptance	Aceptance	Holding	Total Days
Total Closed	157	Total Closed	182	2.82	0.57	1.34	0.53	0.04	0.55	**5.85**
0 Points	9									
Half Point	86	Half Point	43	0.95	0.48	0.59	0.47	0.02	0.35	**2.86**
1 Points	25	1 Points	25	2.60	0.56	1.40	0.36	0.04	0.44	**5.40**
2 Points	12	2 Points	24	5.50	0.42	2.00	0.33	0.00	0.42	**8.67**
3 Points	19	3 Points	57	8.00	1.21	4.47	1.32	0.05	1.95	**17.00**
5 Points	5	5 Points	25	9.20	0.40	2.60	0.20	0.00	0.60	**13.00**
8 Points	1	8 Points	8	25.00	0.00	2.00	0.00	0.00	0.00	**27.00**
12 Points	0	12 Points	0							

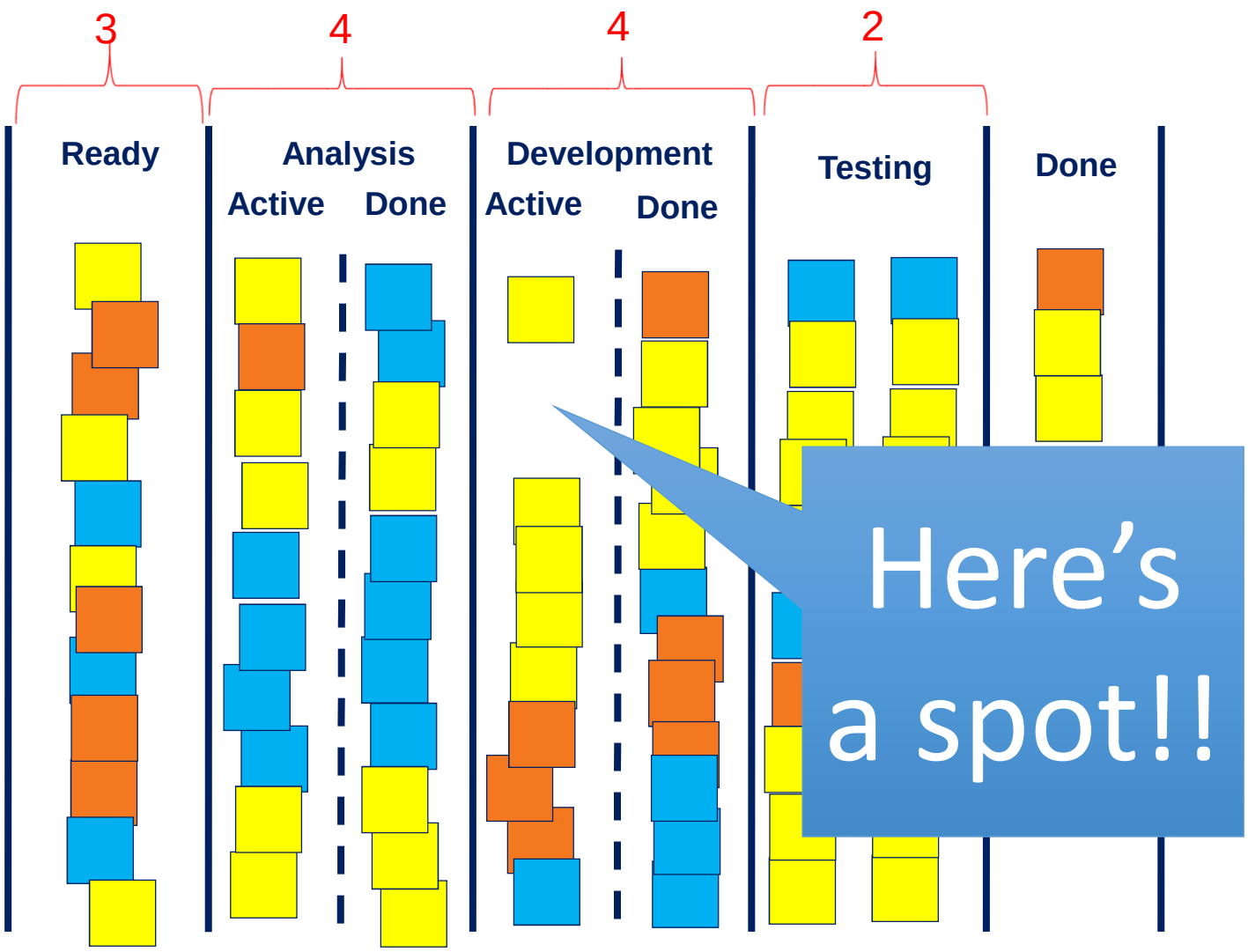
Why do we even bother with Story Points?

So what does matter when it comes to predicting how long it takes an item to complete?

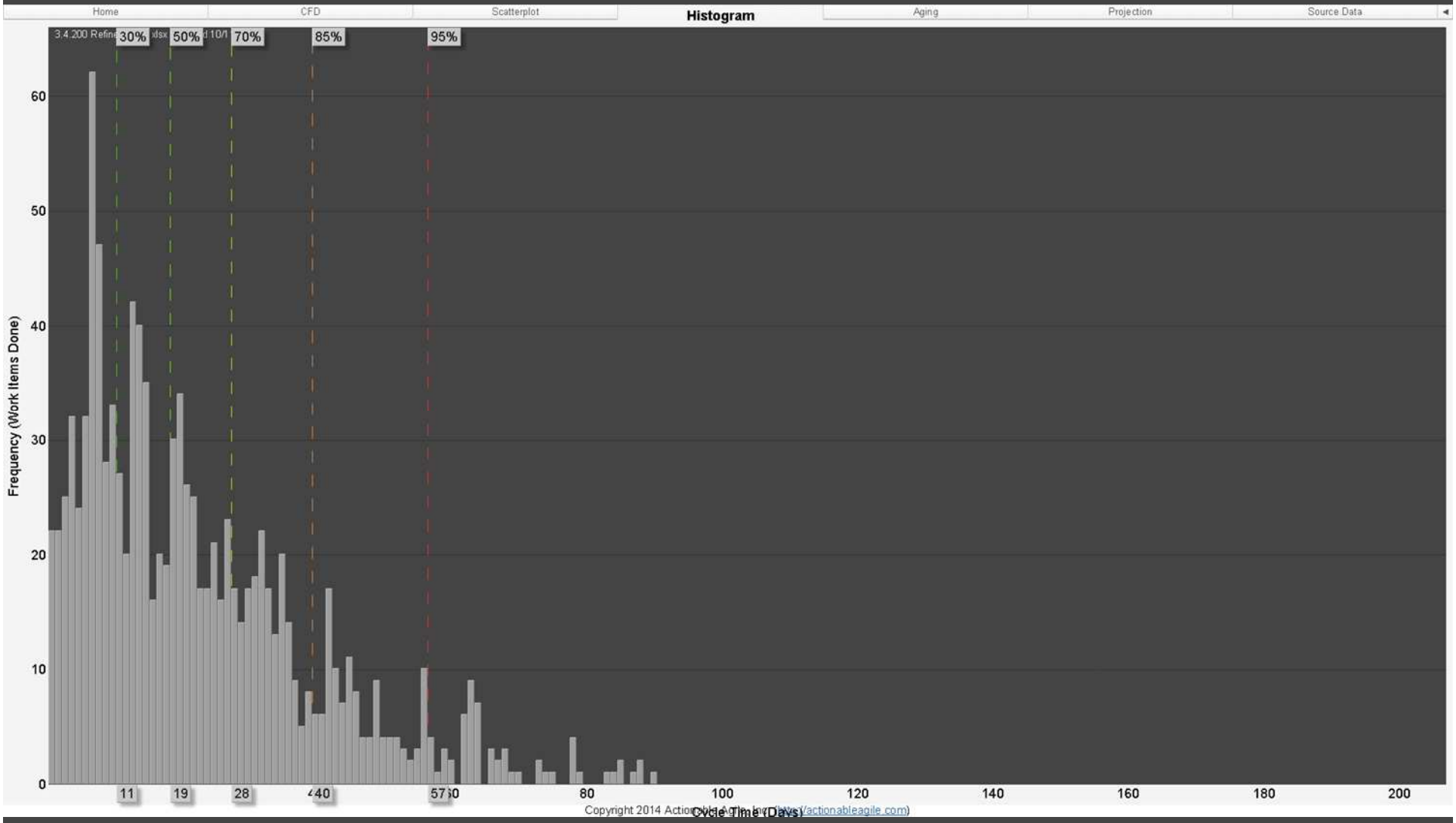
How long does it take you to get to work in the morning?

“It depends…”

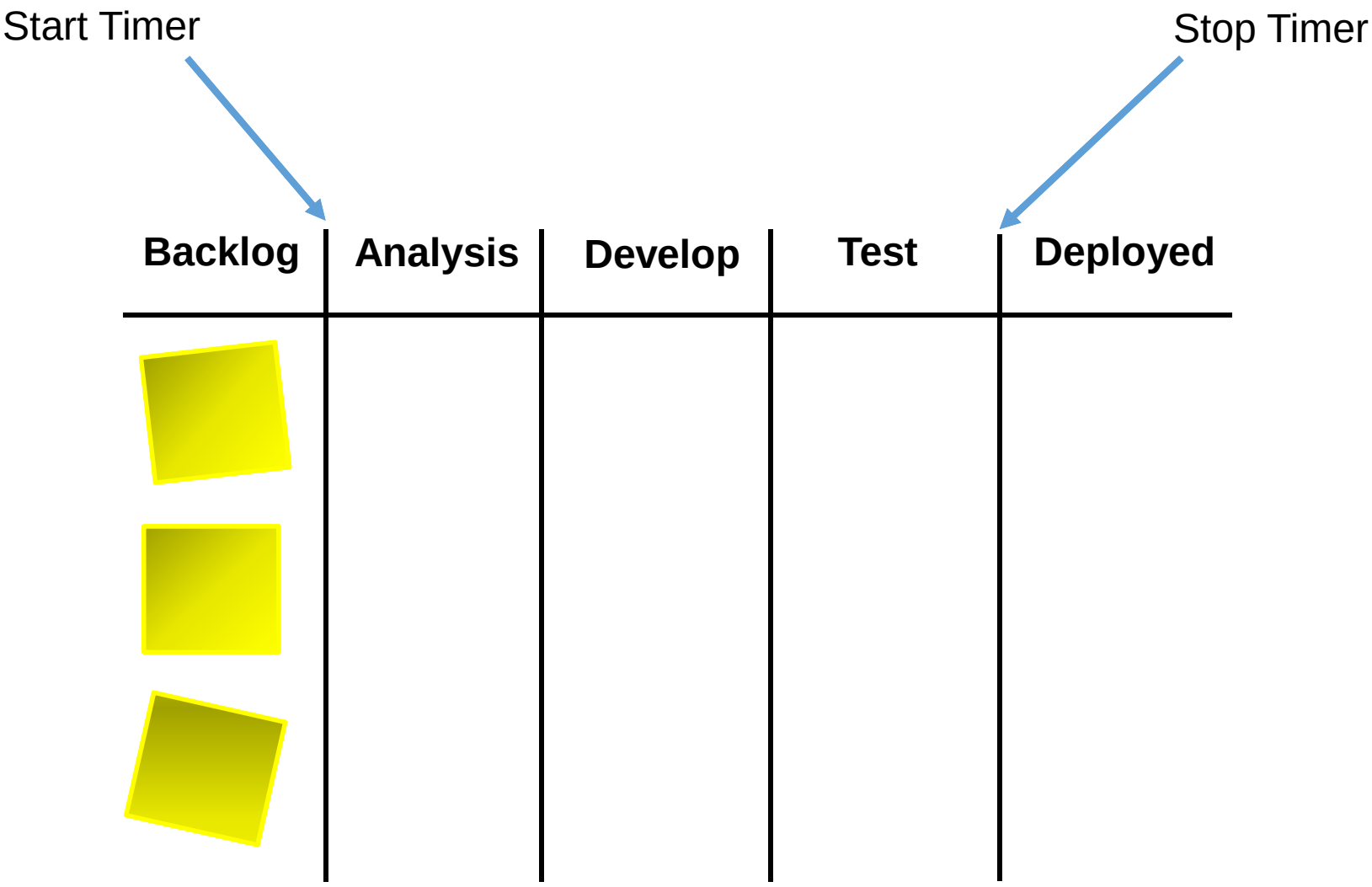

Here's
a spot!!

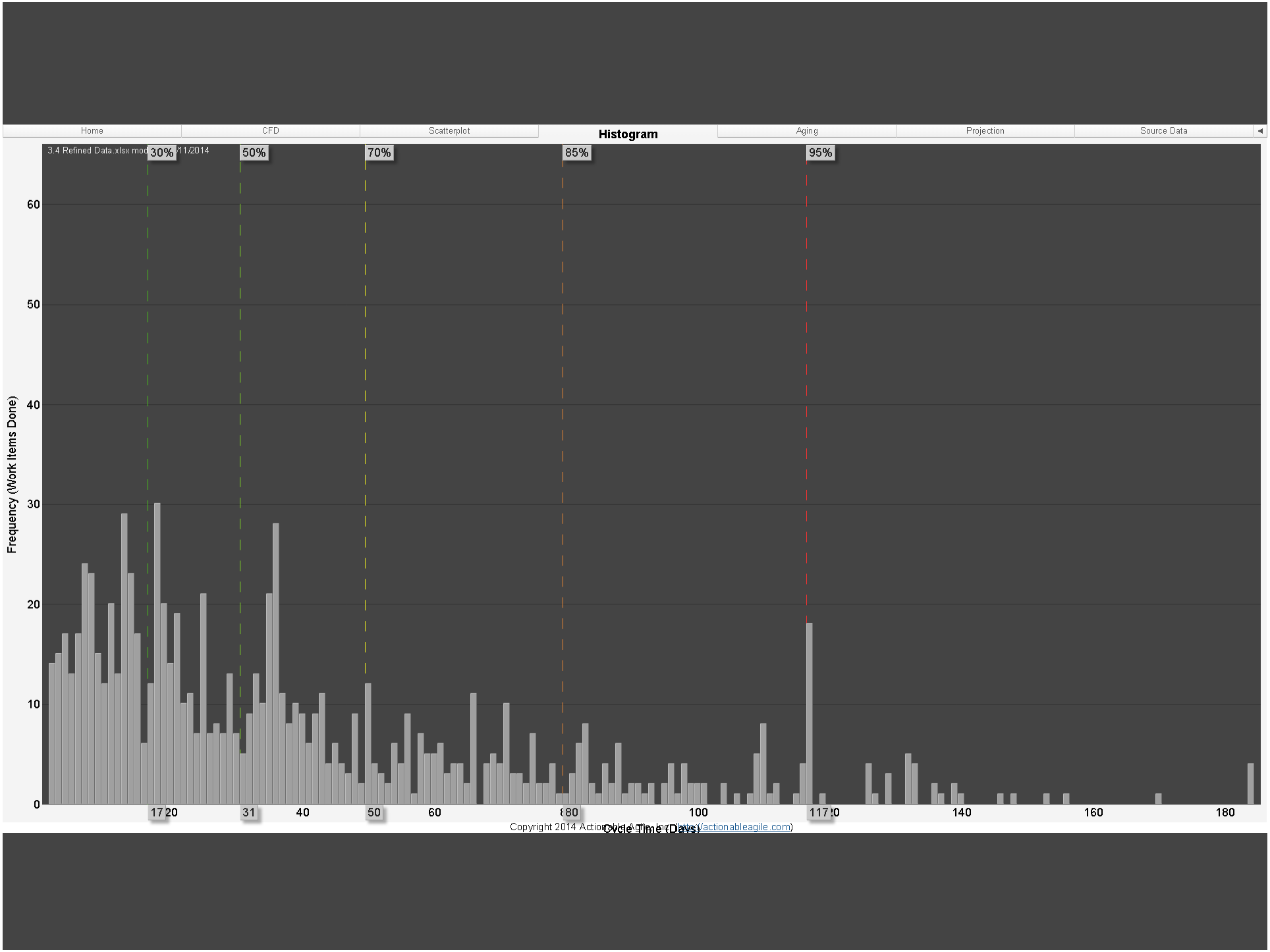

3
4
4
2
Ready
Analysis
Active
Done
Development
Active
Done
Testing
Done
Here's a spot!!

Try an experiment for me


Home
CFD
Scatterplot
Histogram
Aging
Projection
Source Data
30%
50%
70%
85%
95%
Frequency (Work Items Done)
Cycle Time (Days)
0
10
20
30
40
50
60
11
19
28
80
100
120
140
160
180
200
Copyright 2014 Actionable Agile Inc (http://actionableagile.com)

Try the same thing for your process


Start Timer
Stop Timer
Backlog
Analysis
Develop
Test
Deployed


Home
CFD
Scatterplot
Histogram
Aging
Projection
Source Data
3.4 Refined Data.xlsx mod
/11/2014
30%
50%
70%
85%
95%
Frequency (Work Items Done)
0
10
20
30
40
50
60
17
20
31
40
50
60
80
100
117
140
160
180
Cycle Time (Days)
Copyright 2014 Actionable Agile Academy (http://actionableagile.com)

What we've just done is measure the flow metric of

Cycle Time

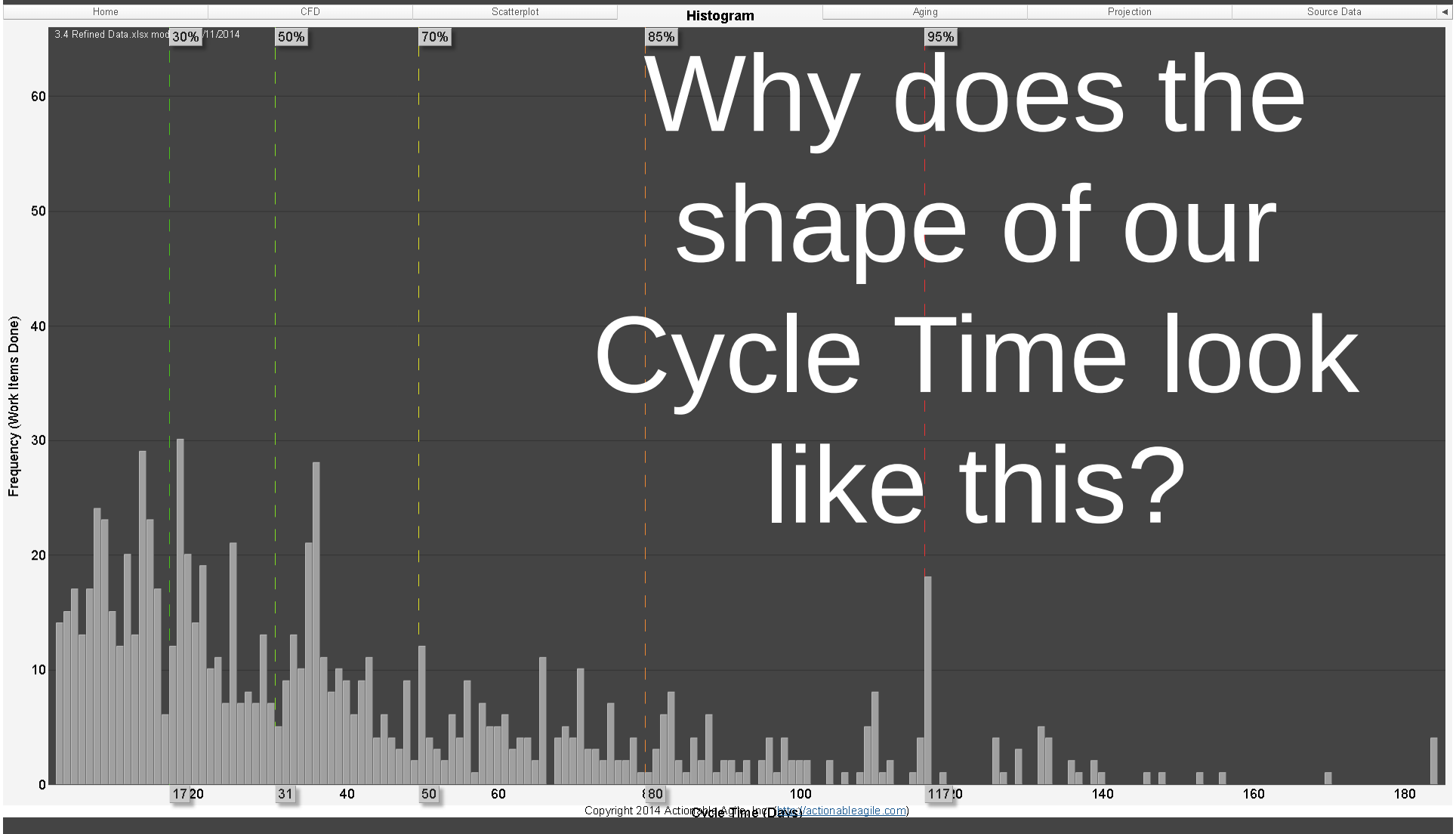
Cycle Time is the amount of

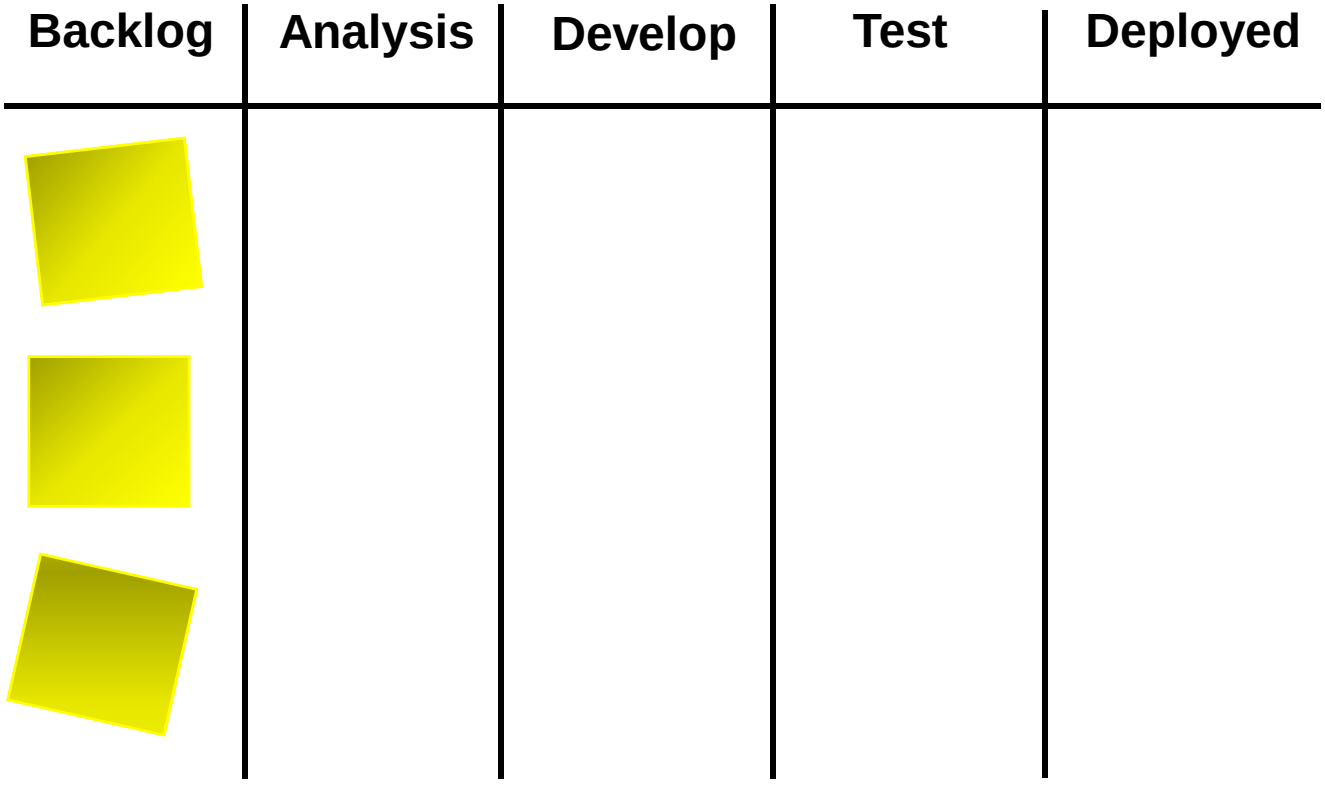
elapsed time

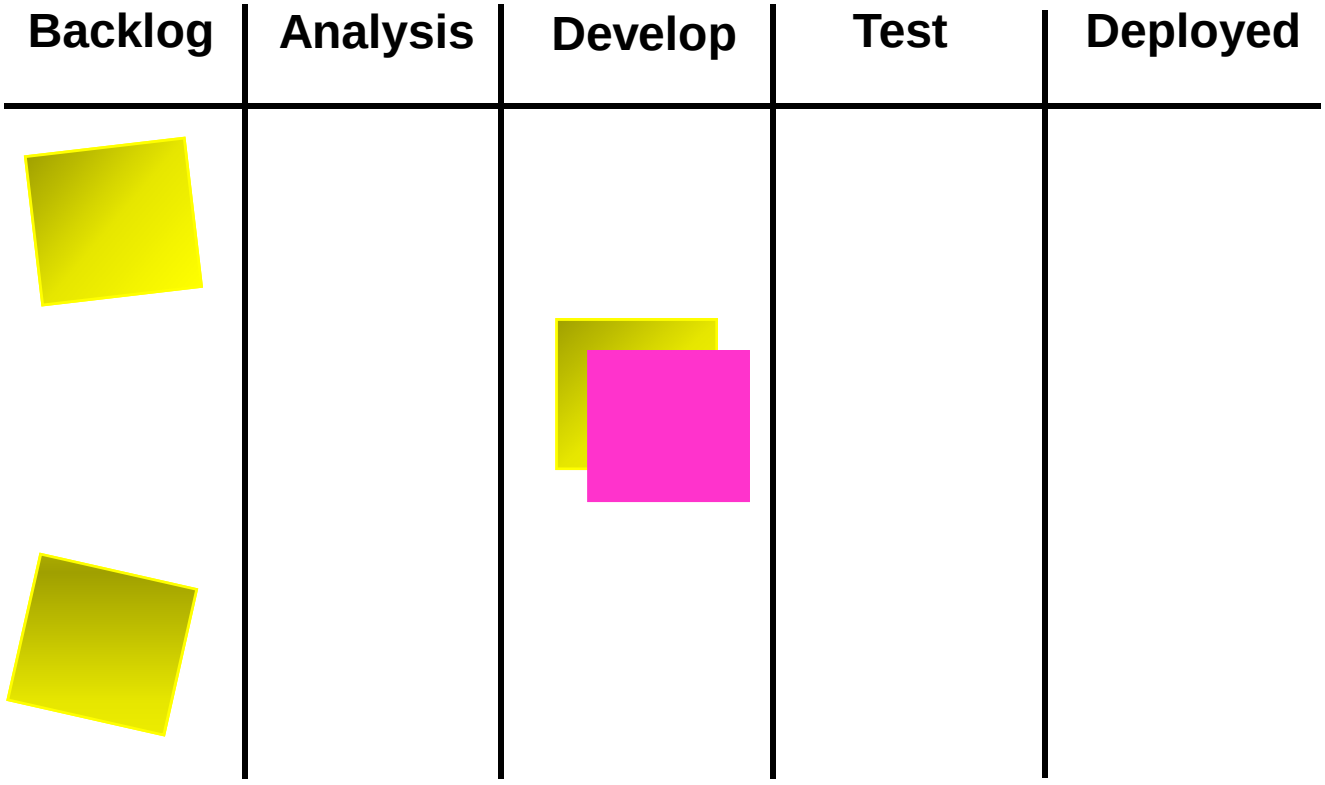
it takes for a given work item to complete

“When will it be done?”
for a single items is best
answered by
the flow metric of
Cycle Time

Why Size Doesn’t Matter Reason #1 : Measuring the Wrong Thing


Home
CFD
Scatterplot
Histogram
Aging
Projection
Source Data
3.4 Refined Data.xlsx mod
/11/2014
30%
50%
70%
85%
95%
Why does the shape of our Cycle Time look like this?
Frequency (Work Items Done)
0
10
20
30
40
50
60
17 20
31
40
50
60
80
100
117 0
140
160
180
Cycle Time (Days)
Copyright 2014 Actionable Agile Academy (www.actionableagile.com)


Backlog
Analysis
Develop
Test
Deployed


Backlog
Analysis
Develop
Test
Deployed

Backlog	Design	Develop	Test	Deployed

1

2

3

Will Item #3 end up…

Exactly Here?

Exactly Here?

Or Exactly Here?

We can’t think deterministically.
We need to think probabilistically.

What does it mean to think probabilistically?

Thinking probabilistically means acknowledging there is more than one possible future outcome

Think of Cycle Time as a shape and not a number

When it comes to predictability, focus on how to affect the shape of your Cycle Time distribution

What factors affect shape?

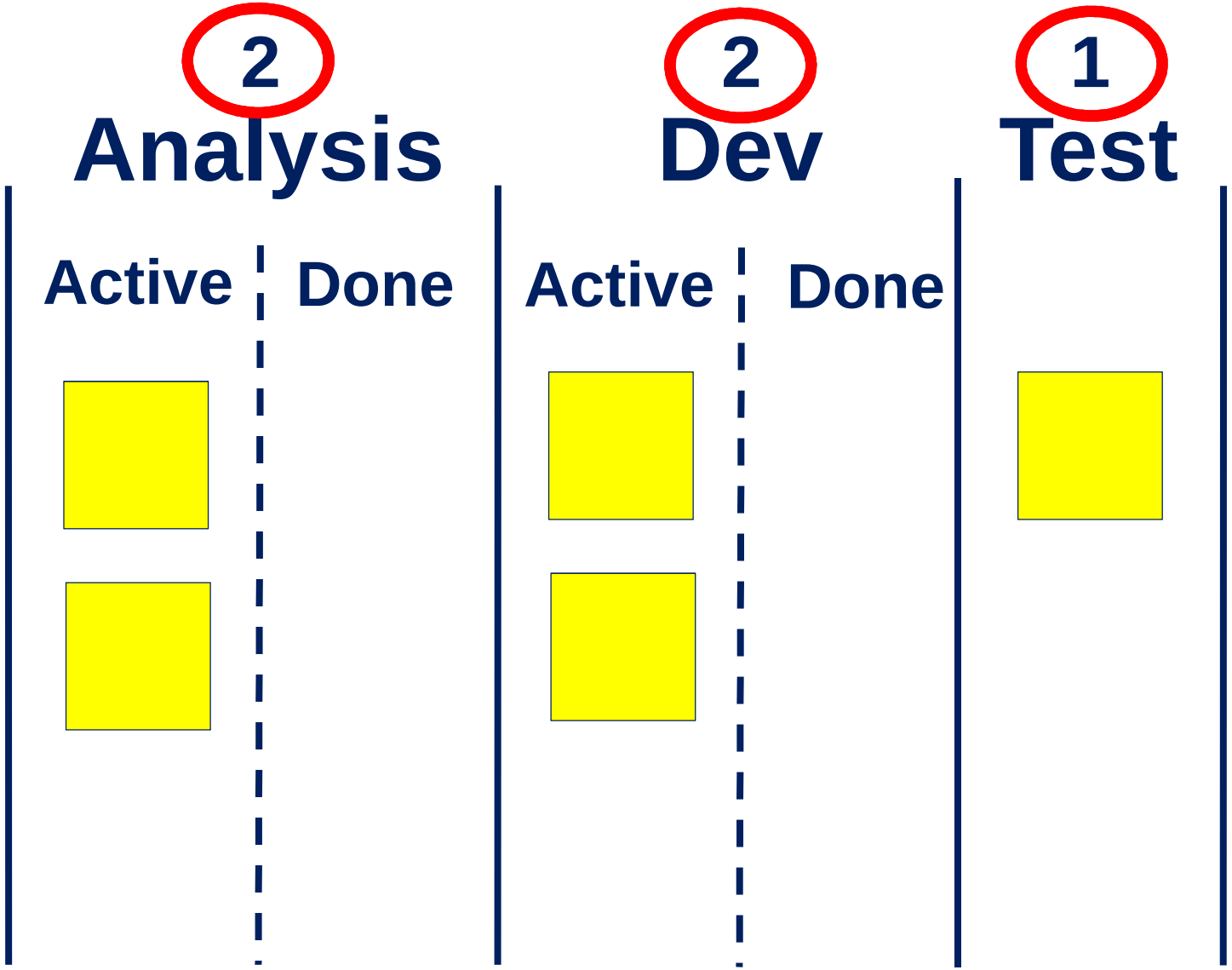

WIP

$$\text{Average Cycle Time} = \frac{\text{Average Work in Progress}}{\text{Average Throughput}}$$

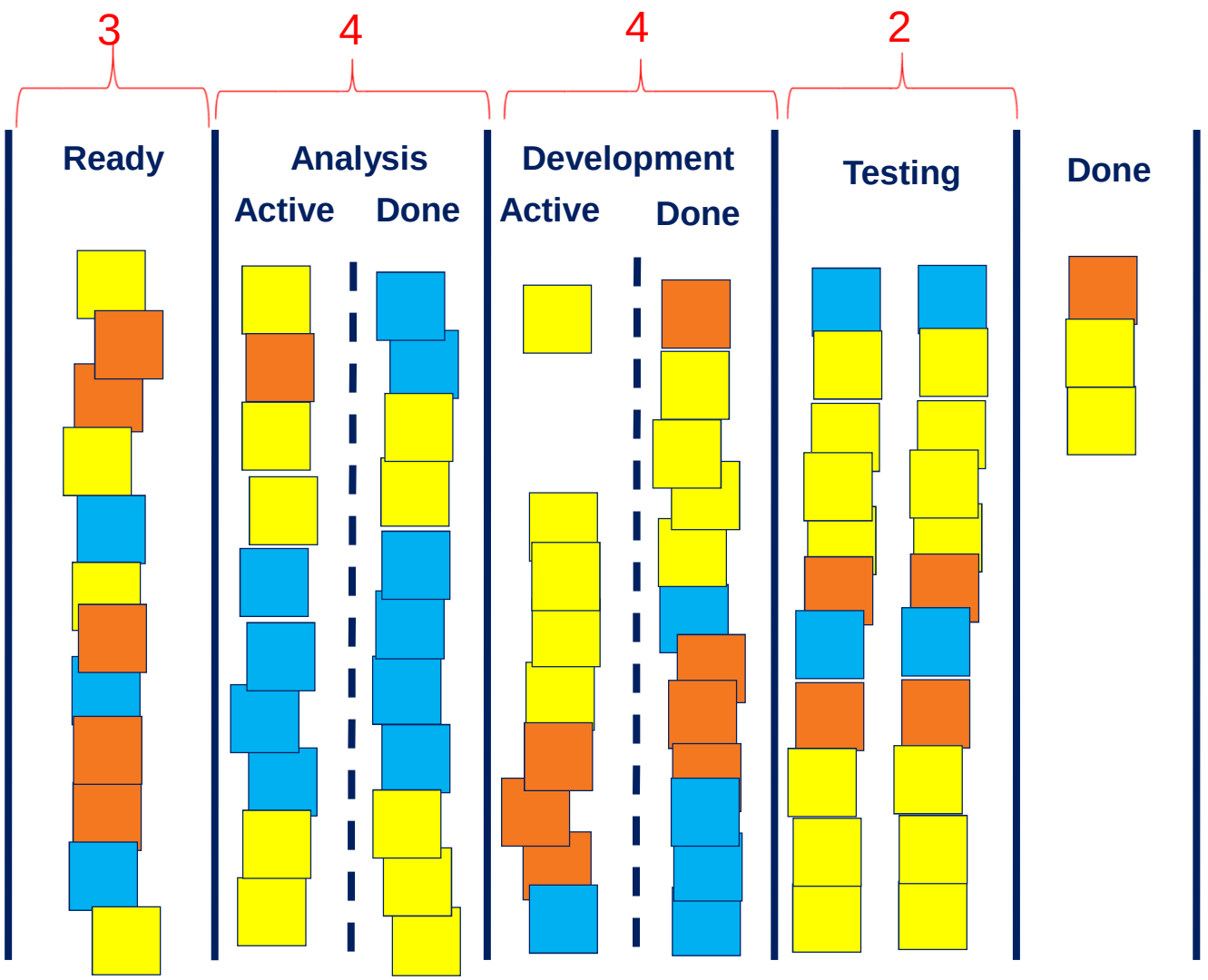
Cycle Time = how long it takes one item to go through the process

Work in Progress = how many items are in the process at any time

Throughput = how many items are produced per unit of time


2
Analysis
Active
Done
2
Dev
Active
Done
1
Test

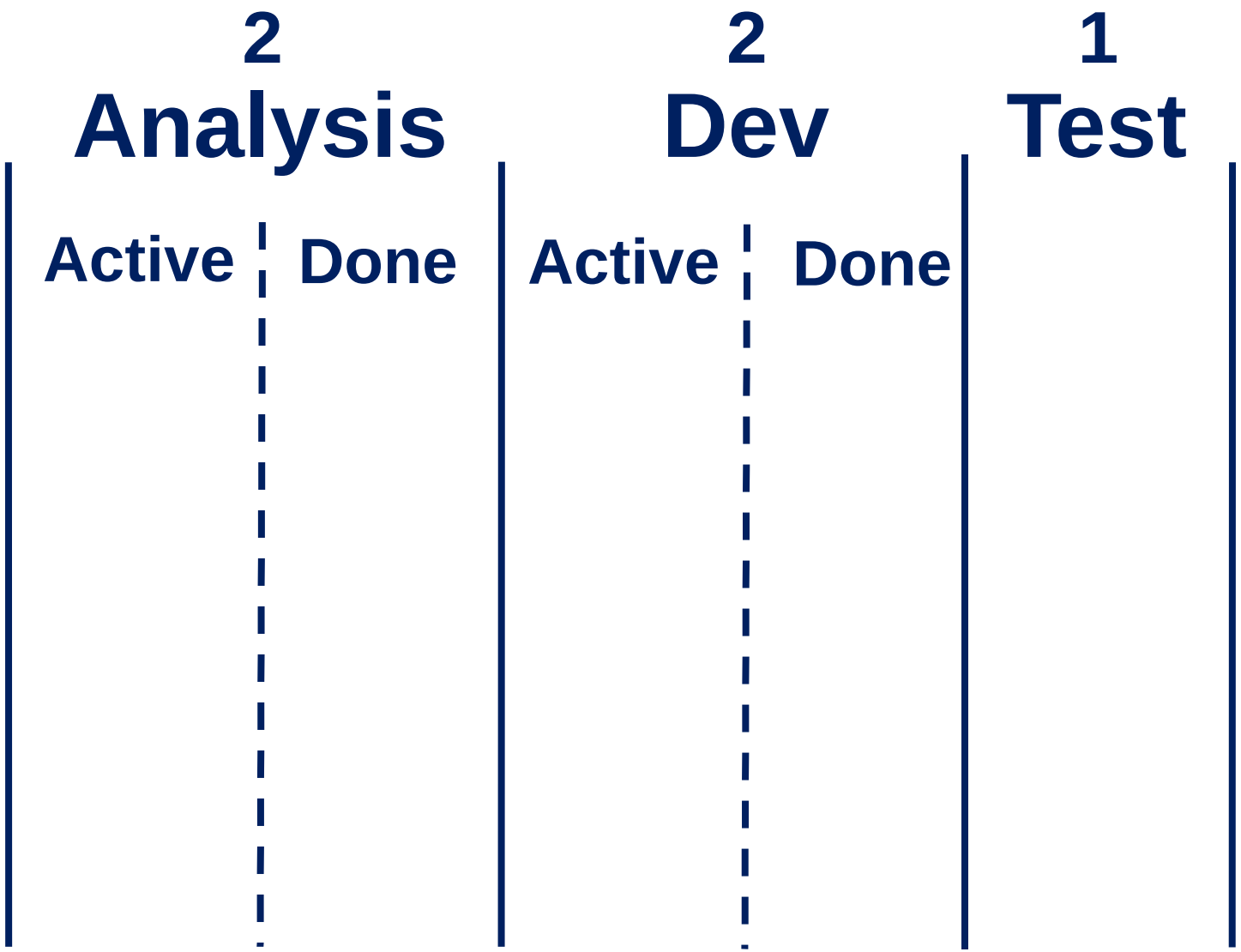
Why Size Doesn’t Matter
Reason #2 :
Too Much WIP


3
4
4
2
Ready
Analysis
Active
Done
Development
Active
Done
Testing
Done

What else?

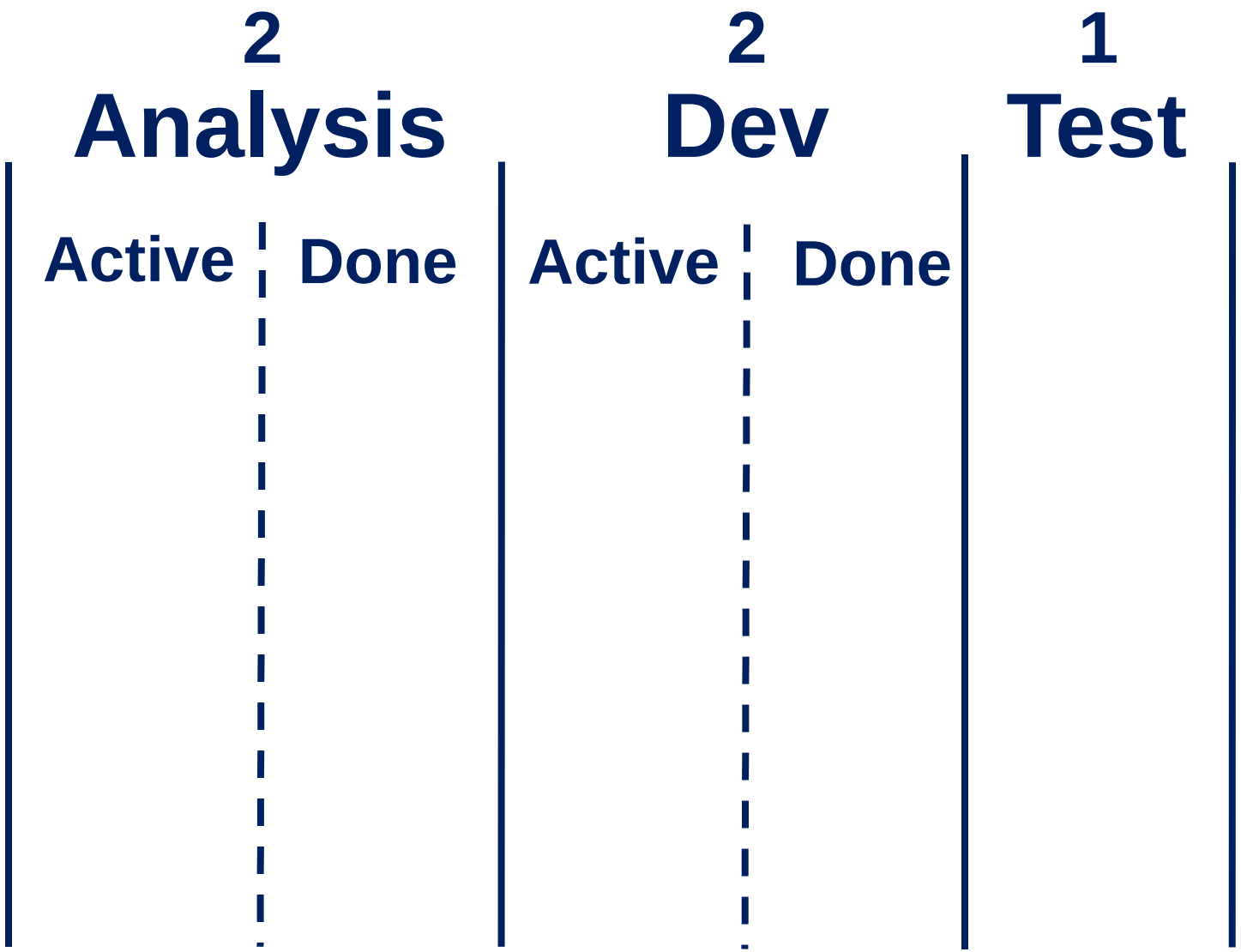
Let’s try another experiment…

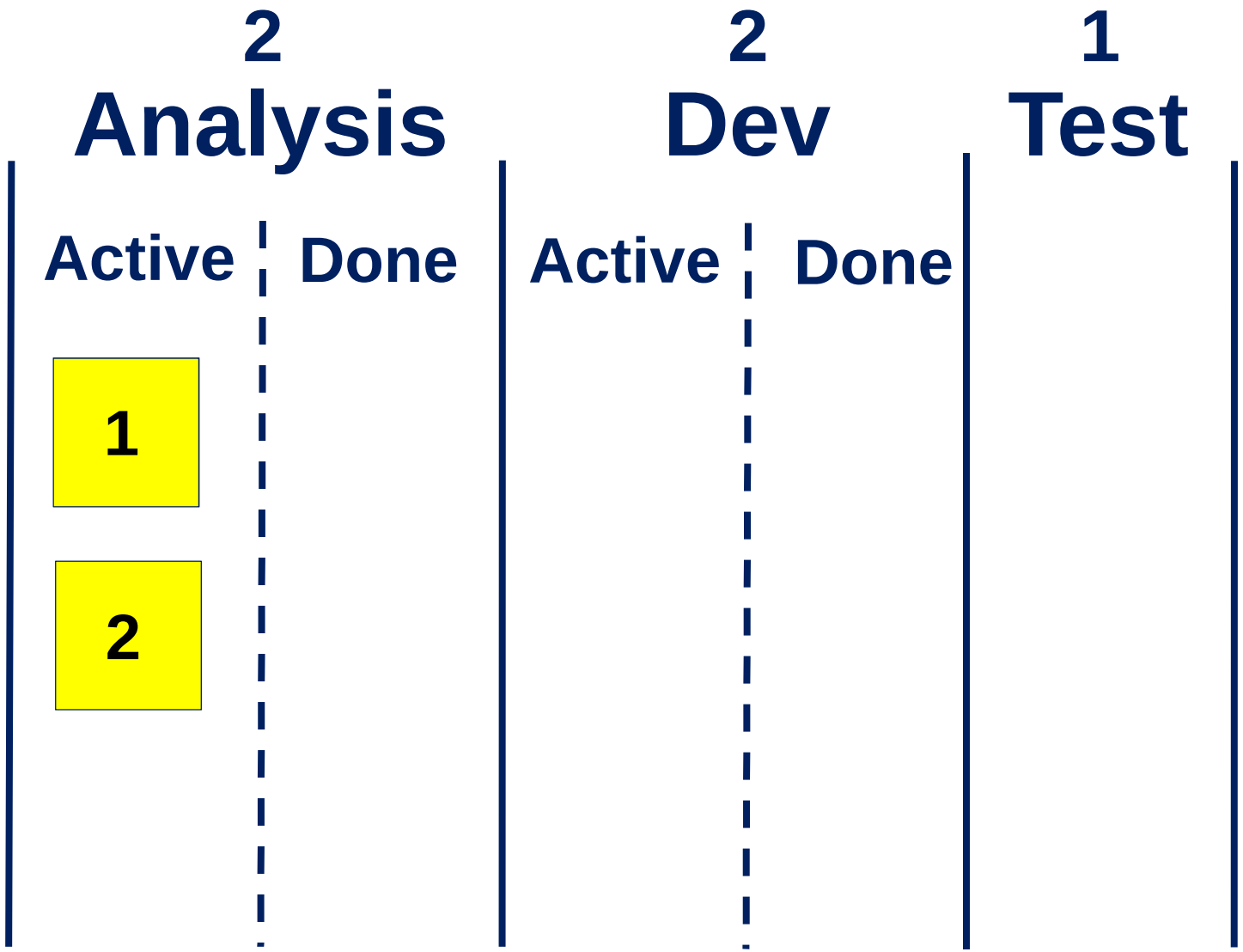
What if we could control WIP and make all items the same “size”?


2
Analysis
Active
Done
2
Dev
Active
Done
1
Test

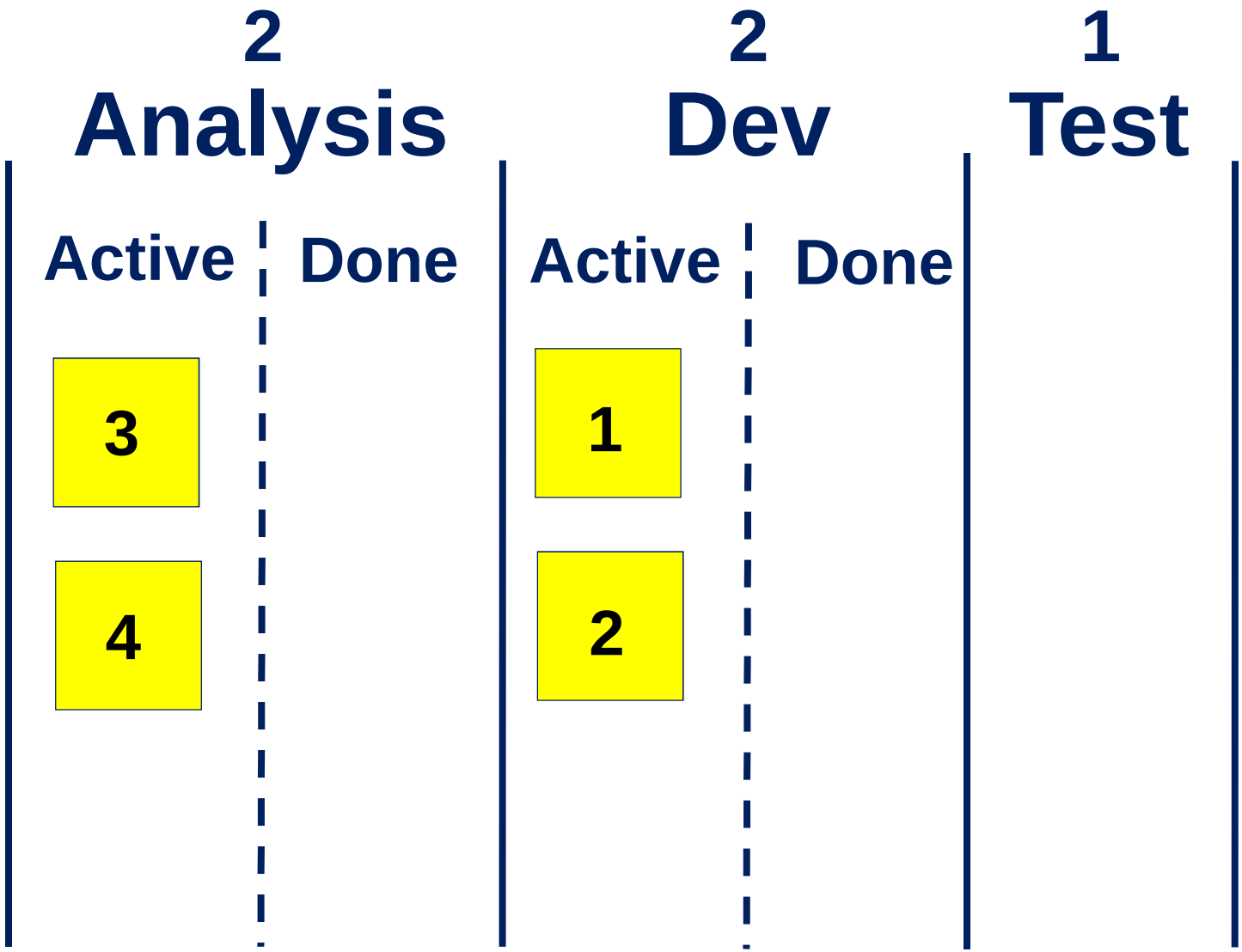
Simulation Setup

- 50 Items in the backlog
- All work items need exactly 10 days of active work in each stage
- No other blocking events or added items

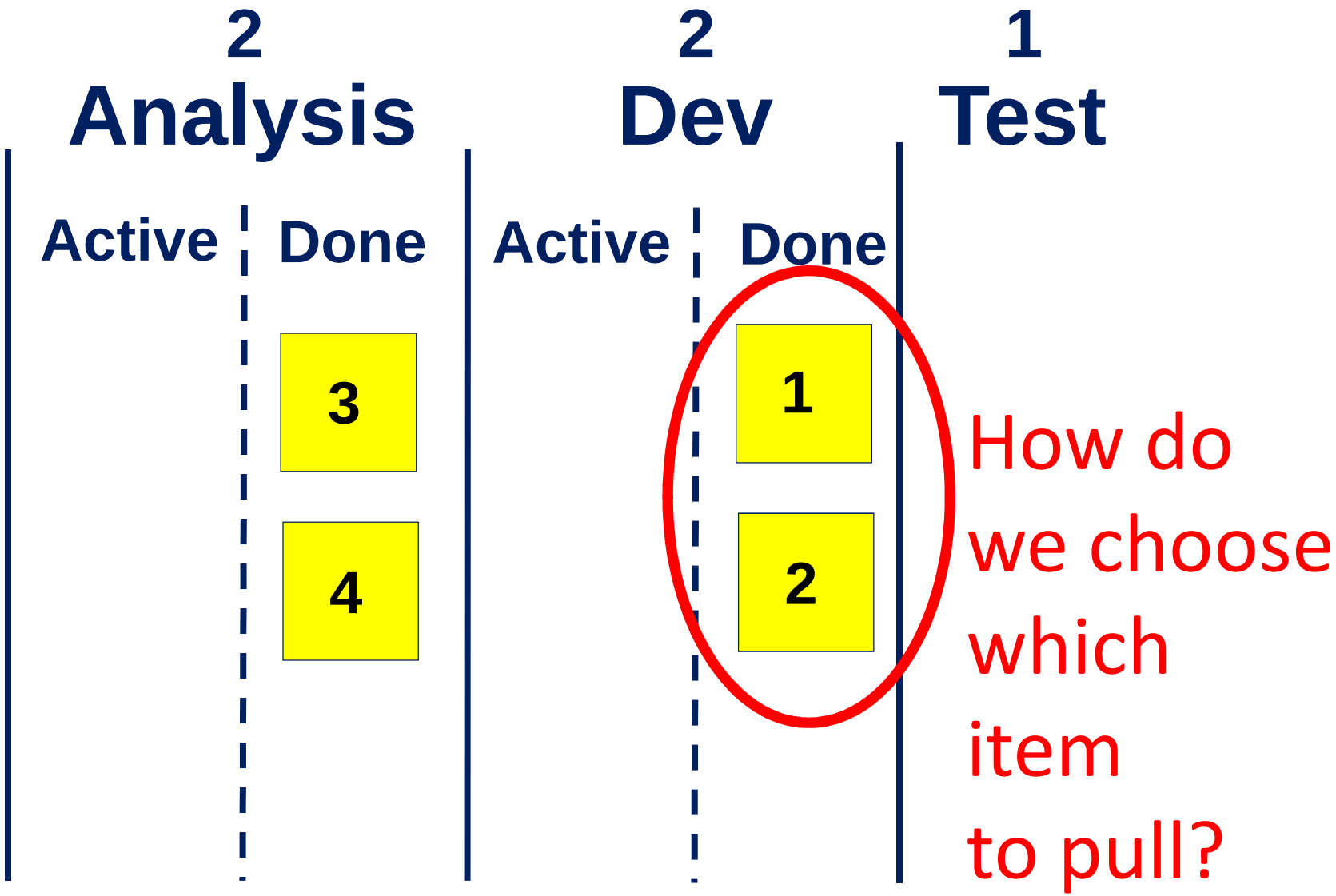

2
Analysis
Active
Done
2
Dev
Active
Done
1
Test



At the end of day 1



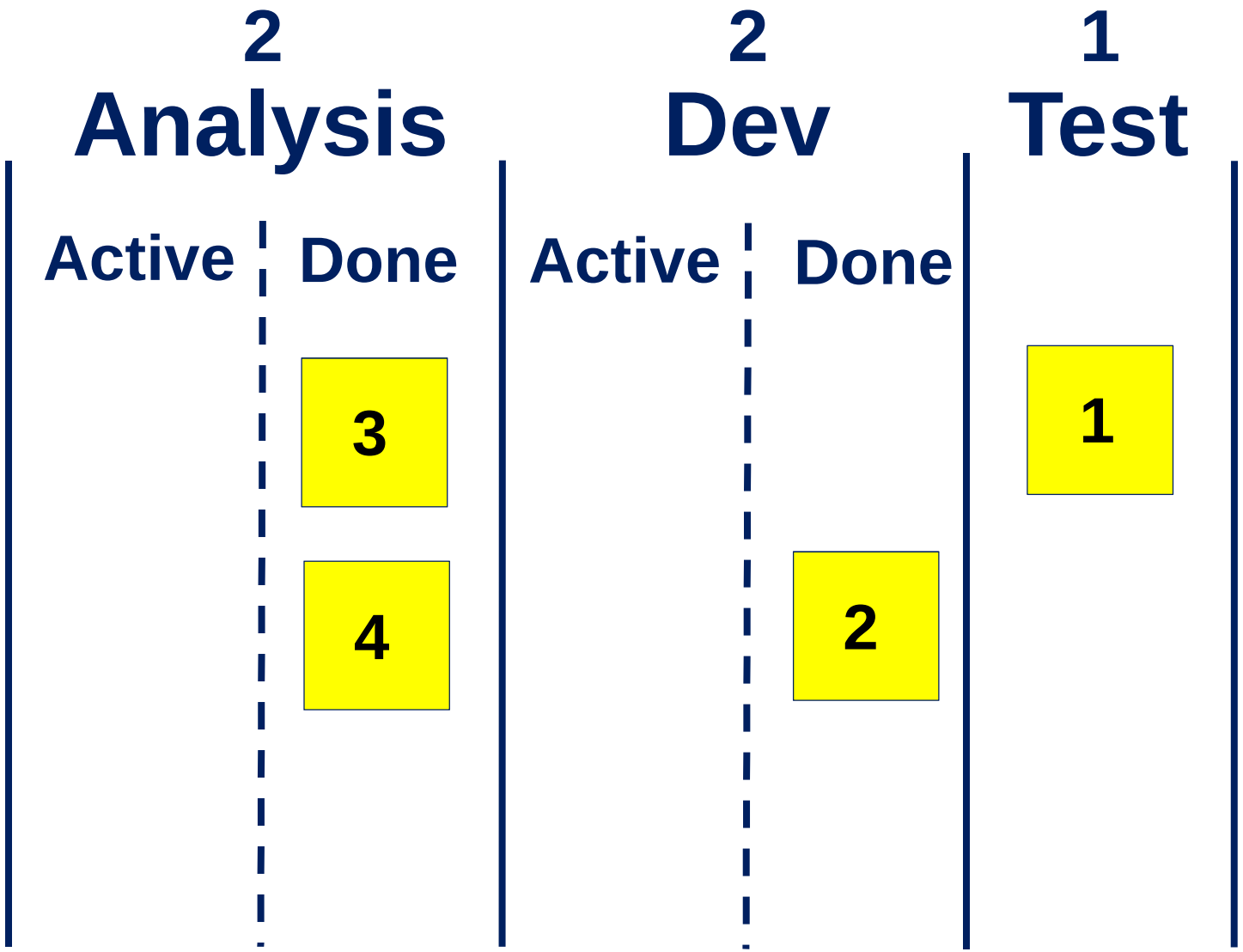
At the end of day 11



At the end of day 20

Baseline Case

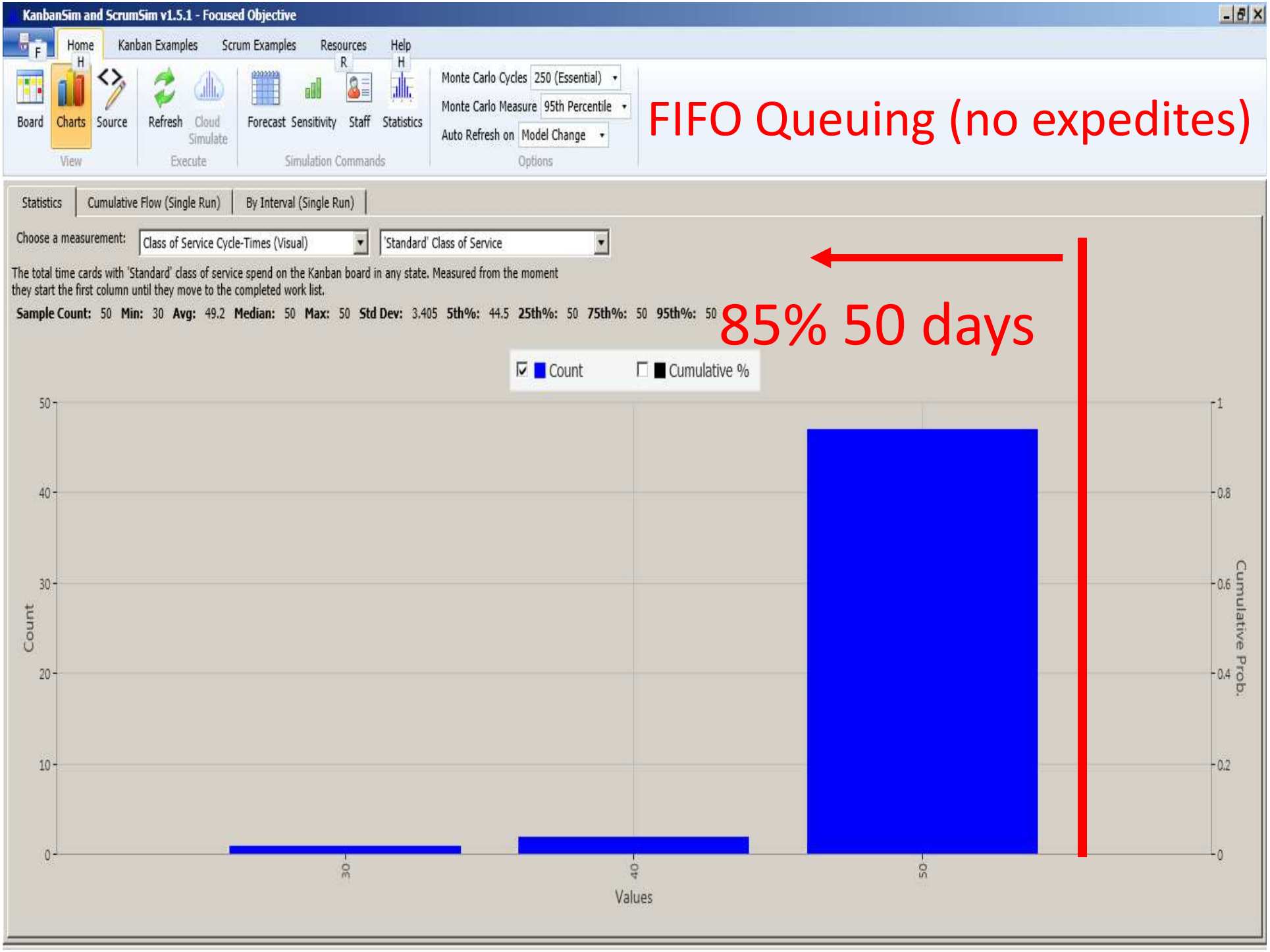
Items are worked through the process in a strict FIFO queuing manner.



At the beginning of day 21

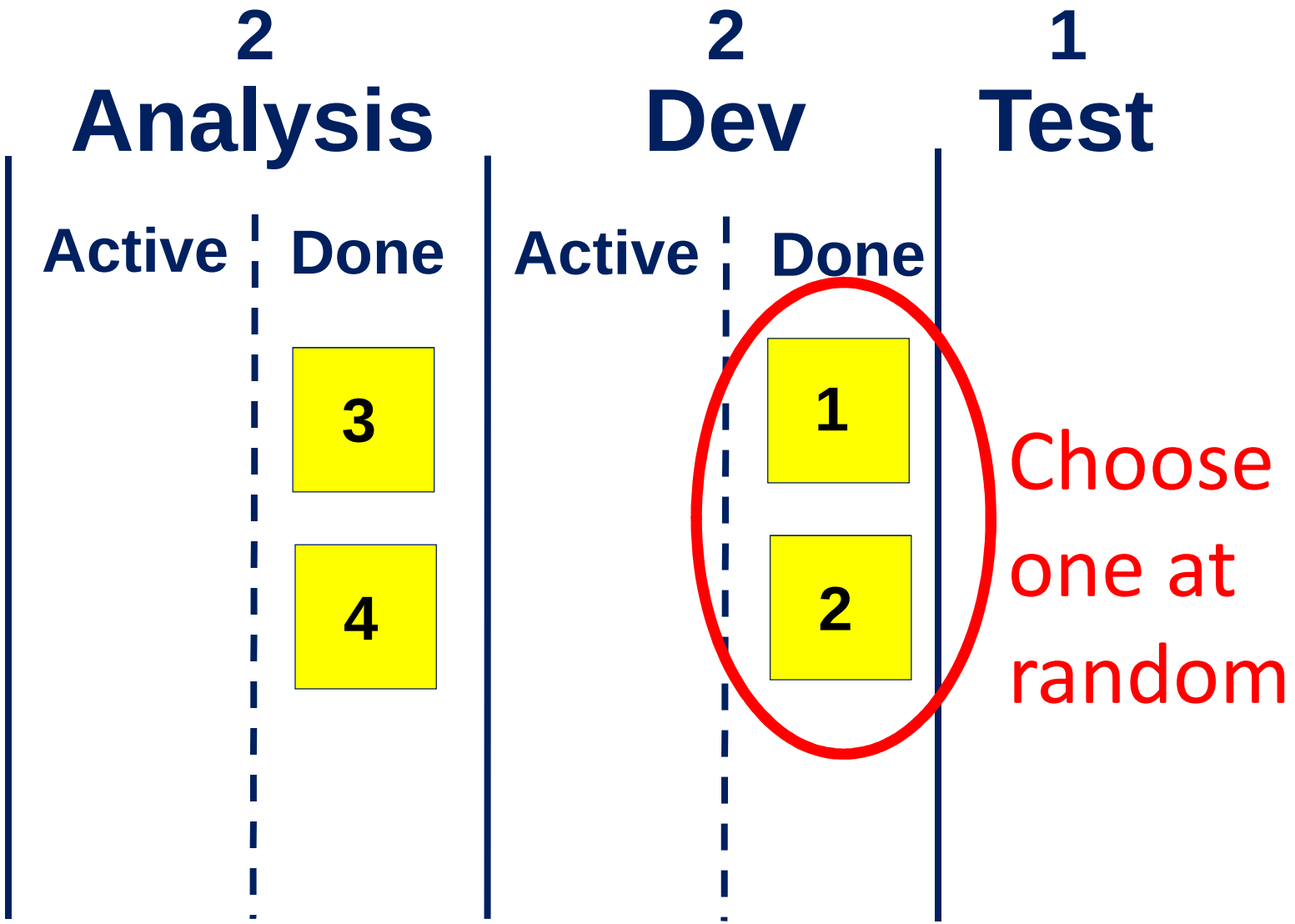
After 50 simulations, what do you think our expected Cycle Time will be at the 85th percentile?

(If you want to put a number on Cycle Time you have to give an associated confidence)

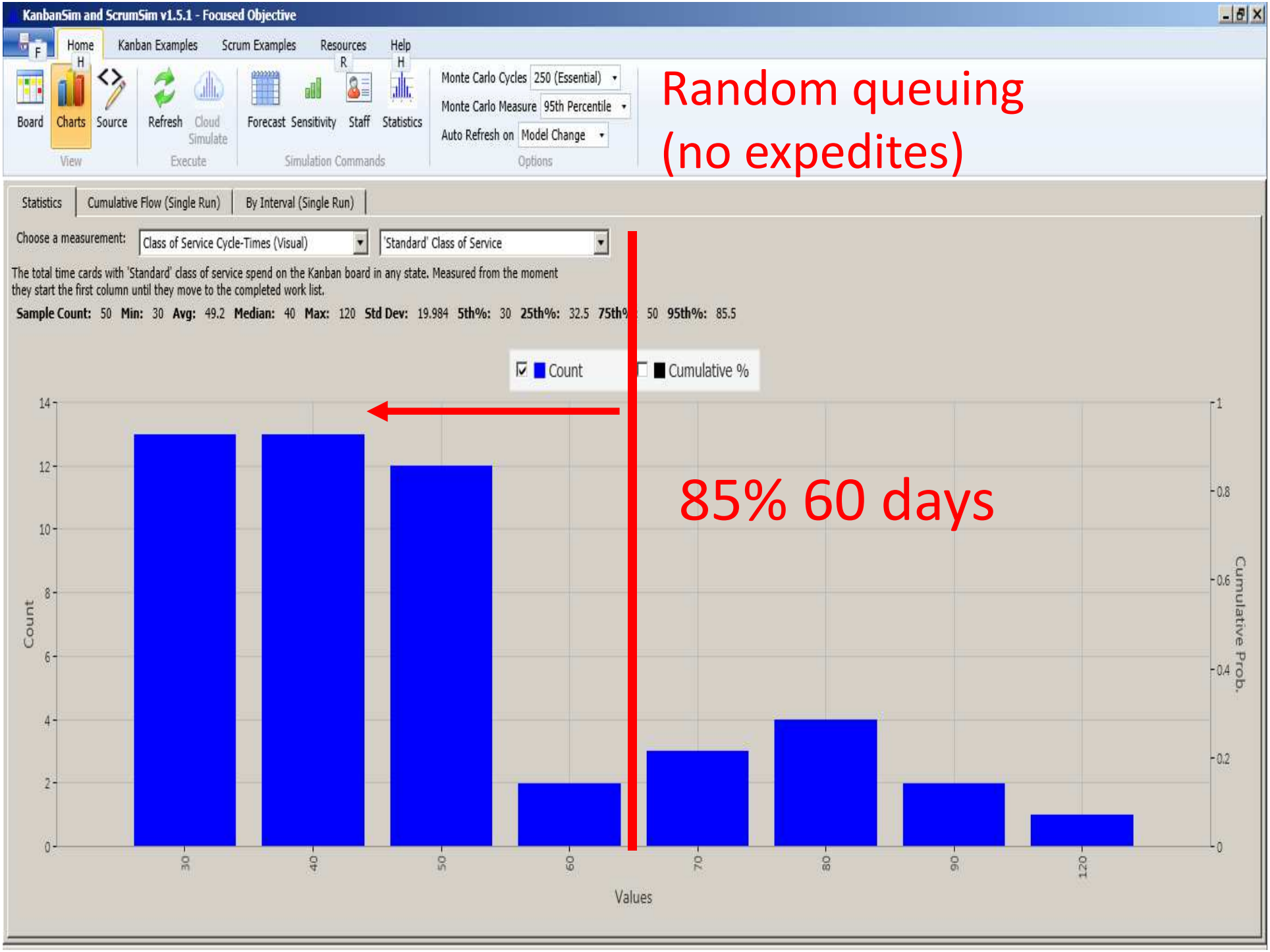

KanbanSim and ScrumSim v1.5.1 - Focused Objective
Home
Kanban Examples
Scrum Examples
Resources
Help
Board
Charts
Source
Refresh
Cloud Simulate
Forecast
Sensitivity
Staff
Statistics
View
Execute
Simulation Commands
Monte Carlo Cycles 250 (Essential)
Monte Carlo Measure 95th Percentile
Auto Refresh on Model Change
Options
FIFO Queuing (no expedites)
Statistics
Cumulative Flow (Single Run)
By Interval (Single Run)
Choose a measurement:
Class of Service Cycle-Times (Visual)
'Standard' Class of Service
The total time cards with 'Standard' class of service spend on the Kanban board in any state. Measured from the moment they start the first column until they move to the completed work list.
Sample Count: 50 Min: 30 Avg: 49.2 Median: 50 Max: 50 Std Dev: 3.405 5th%: 44.5 25th%: 50 75th%: 50 95th%: 50
85% 50 days
Count
Cumulative %
Count
Cumulative Prob.
Values
0
10
20
30
40
50
0
0.2
0.4
0.6
0.8
1
30
40
50

Some Expedites are Introduced

Baseline extended to use random queuing instead of strict FIFO queuing.


2
Analysis
Active
Done
3
4
2
Dev
Active
Done
1
2
1
Test
Choose one at random

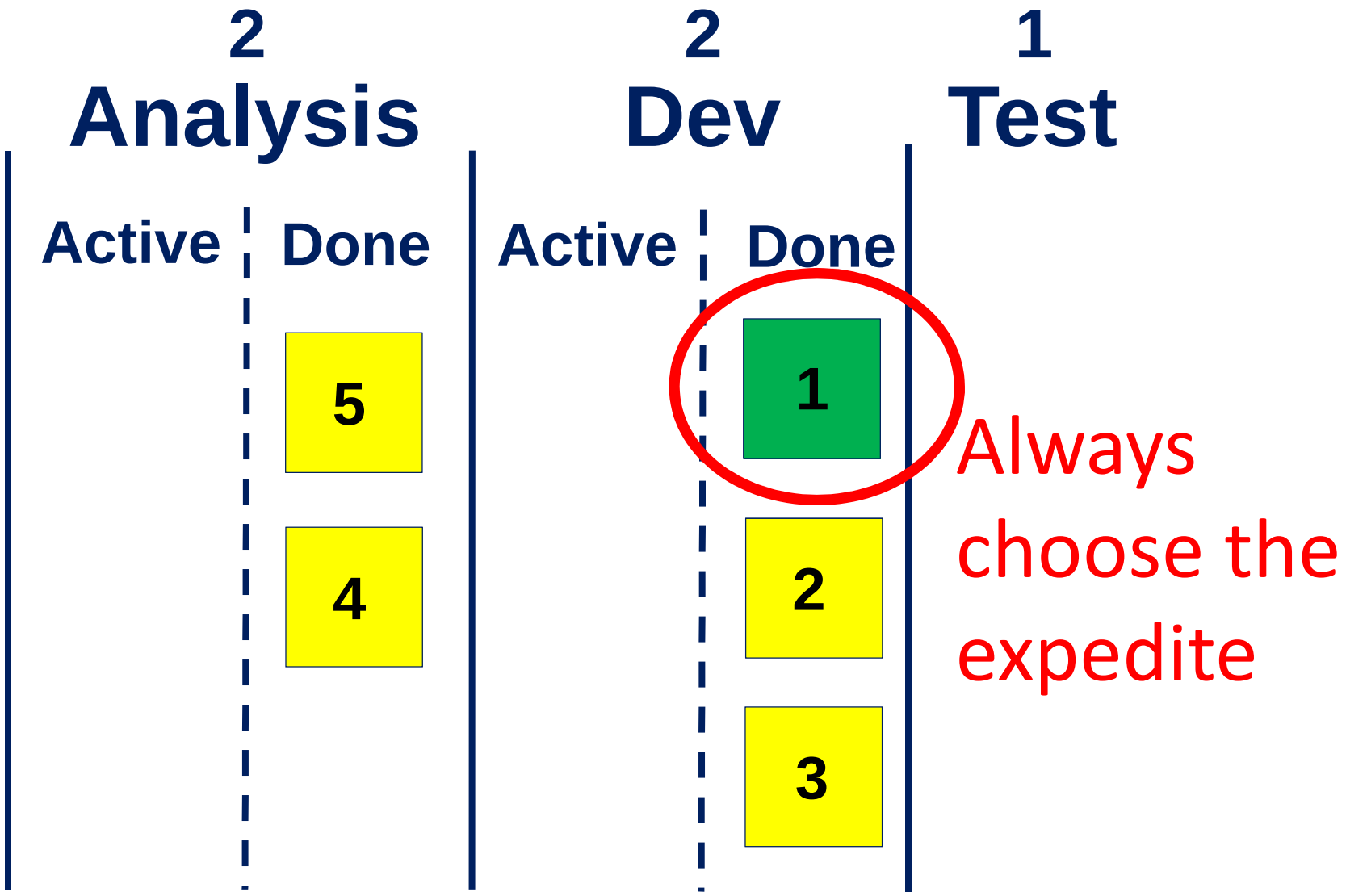
What will this do to Cycle Time? To variability?
To predictability?


KanbanSim and ScrumSim v1.5.1 - Focused Objective
Home
Kanban Examples
Scrum Examples
Resources
Help
Board
Charts
Source
Refresh
Cloud Simulate
Forecast
Sensitivity
Staff
Statistics
Monte Carlo Cycles 250 (Essential)
Monte Carlo Measure 95th Percentile
Auto Refresh on Model Change
View
Execute
Simulation Commands
Options
Random queuing
(no expedites)
Statistics
Cumulative Flow (Single Run)
By Interval (Single Run)
Choose a measurement:
Class of Service Cycle-Times (Visual)
'Standard' Class of Service
The total time cards with 'Standard' class of service spend on the Kanban board in any state. Measured from the moment they start the first column until they move to the completed work list.
Sample Count: 50 Min: 30 Avg: 49.2 Median: 40 Max: 120 Std Dev: 19.984 5th%: 30 25th%: 32.5 75th% 50 95th%: 85.5
Count
Cumulative %
85% 60 days
Count
Cumulative Prob.
Values

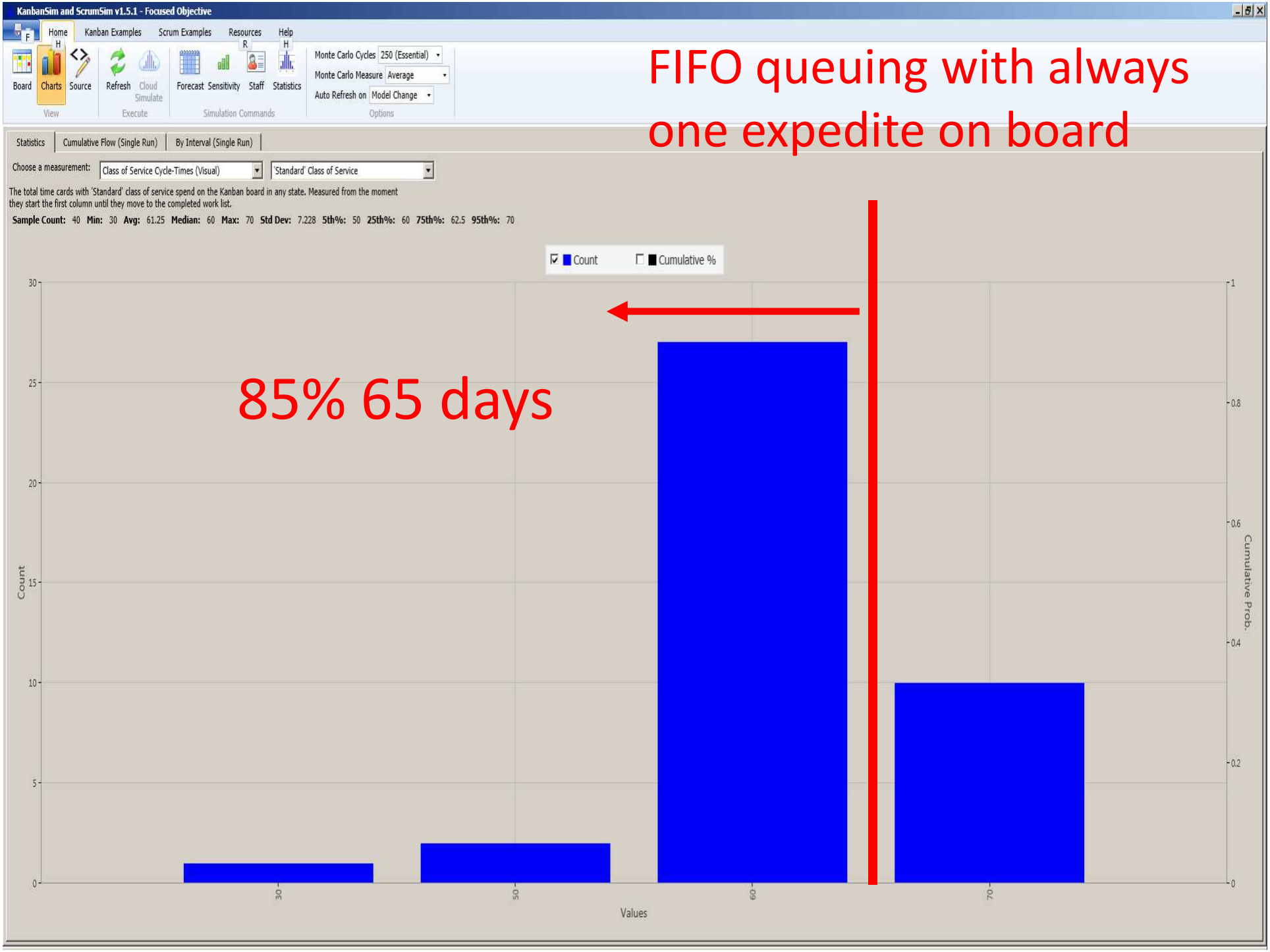
Some Expedites are Introduced

Baseline case extended so that there is always one (and only one) expedite on the board. Strict FIFO queuing is reintroduced.

Expedites can violate WIP Limits and are always pulled in preference to other items


2
Analysis
Active
Done
5
4
2
Dev
Active
Done
1
2
3
1
Test
Always choose the expedite

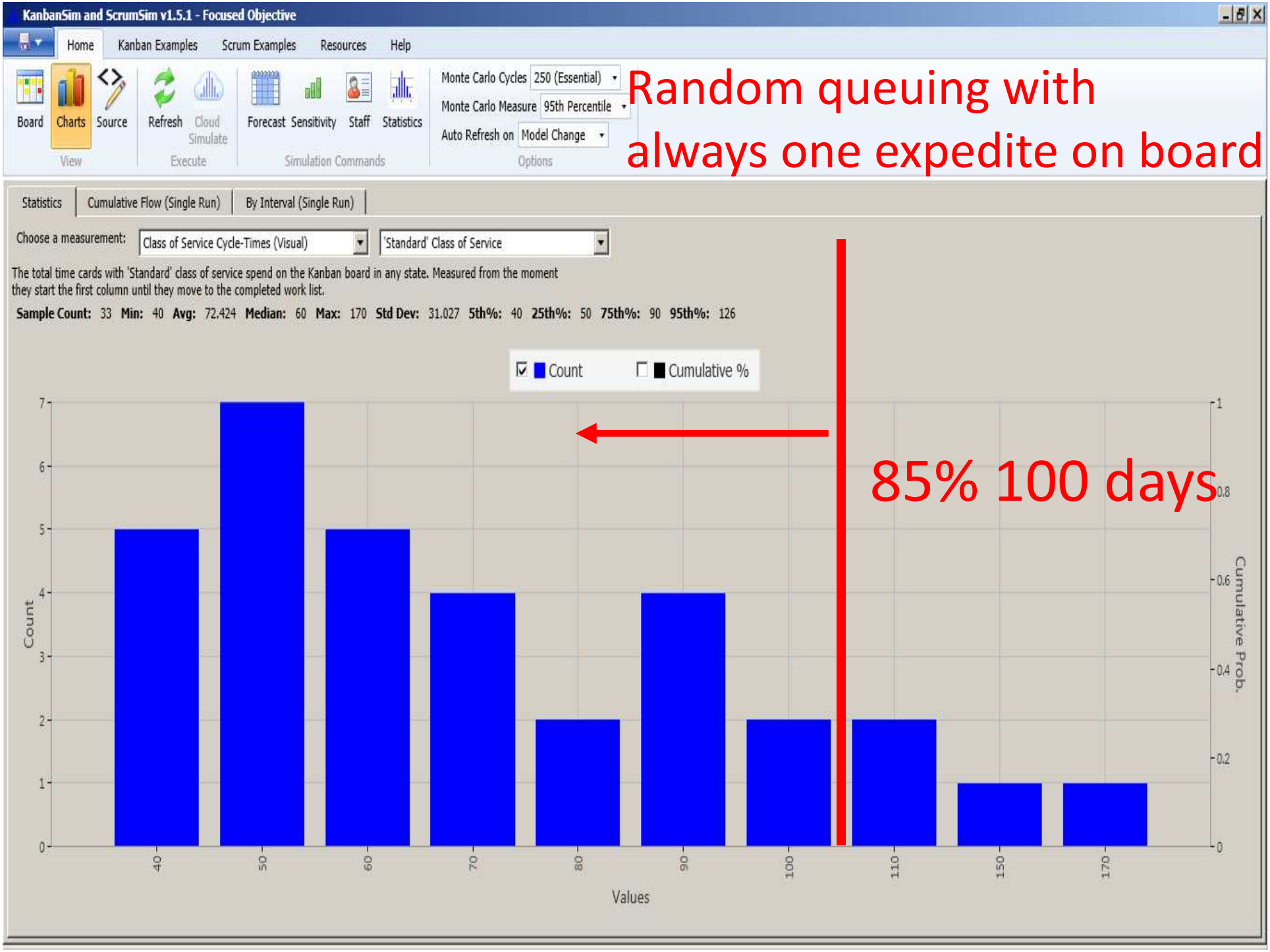
What do you think our expected Cycle Time will be at the 85th percentile for the non-escalated items?


KanbanSim and ScrumSim v1.5.1 - Focused Objective
Home
Kanban Examples
Scrum Examples
Resources
Help
Board
Charts
Source
Refresh
Cloud Simulate
Forecast
Sensitivity
Staff
Statistics
Monte Carlo Cycles 250 (Essential)
Monte Carlo Measure Average
Auto Refresh on Model Change
View
Execute
Simulation Commands
Options
FIFO queuing with always one expedite on board
Statistics
Cumulative Flow (Single Run)
By Interval (Single Run)
Choose a measurement:
Class of Service Cycle-Times (Visual)
'Standard' Class of Service
The total time cards with 'Standard' class of service spend on the Kanban board in any state. Measured from the moment they start the first column until they move to the completed work list.
Sample Count: 40 Min: 30 Avg: 61.25 Median: 60 Max: 70 Std Dev: 7.228 5th%: 50 25th%: 60 75th%: 62.5 95th%: 70
Count
Cumulative %
85% 65 days
Count
Cumulative Prob.
Values
30
50
60
70

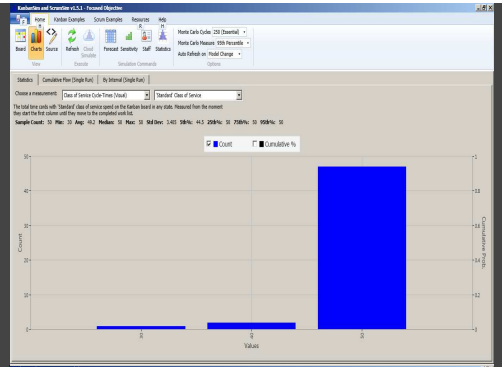
Some Expedites are Introduced

Baseline case extended so that there is always one (and only one) expedite on the board. Random queuing is reintroduced.

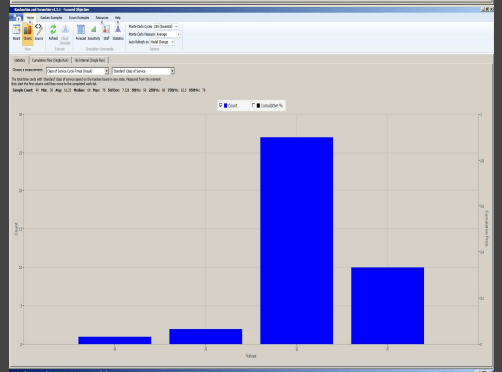
What will this do to standard class Cycle Time? To variability? To predictability?


KanbanSim and ScrumSim v1.5.1 - Focused Objective
Home
Kanban Examples
Scrum Examples
Resources
Help
Board
Charts
Source
Refresh
Cloud Simulate
Forecast
Sensitivity
Staff
Statistics
Monte Carlo Cycles 250 (Essential)
Monte Carlo Measure 95th Percentile
Auto Refresh on Model Change
View
Execute
Simulation Commands
Options
Random queuing with always one expedite on board
Statistics
Cumulative Flow (Single Run)
By Interval (Single Run)
Choose a measurement:
Class of Service Cycle-Times (Visual)
'Standard' Class of Service
The total time cards with 'Standard' class of service spend on the Kanban board in any state. Measured from the moment they start the first column until they move to the completed work list.
Sample Count: 33 Min: 40 Avg: 72.424 Median: 60 Max: 170 Std Dev: 31.027 5th%: 40 25th%: 50 75th%: 90 95th%: 126
Count
Cumulative %
85% 100 days
Count
Cumulative Prob.
Values
40
50
60
70
80
90
100
110
150
170

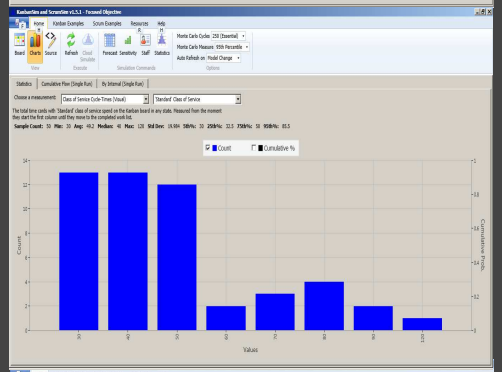
Strict FIFO (no expedites):	50 days
Random Queuing:	60 days
Strict FIFO always one expedite:	65 days
Random Queuing always one expedite:	100 days



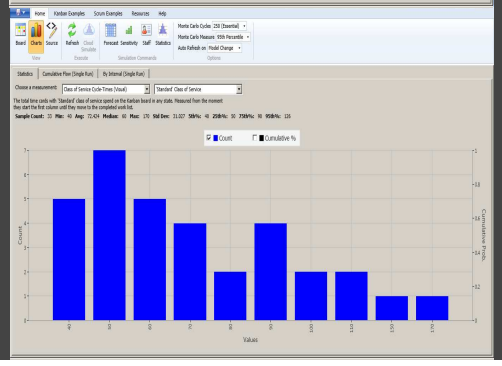
Strict FIFO (no expedites)



Strict FIFO always one expedite



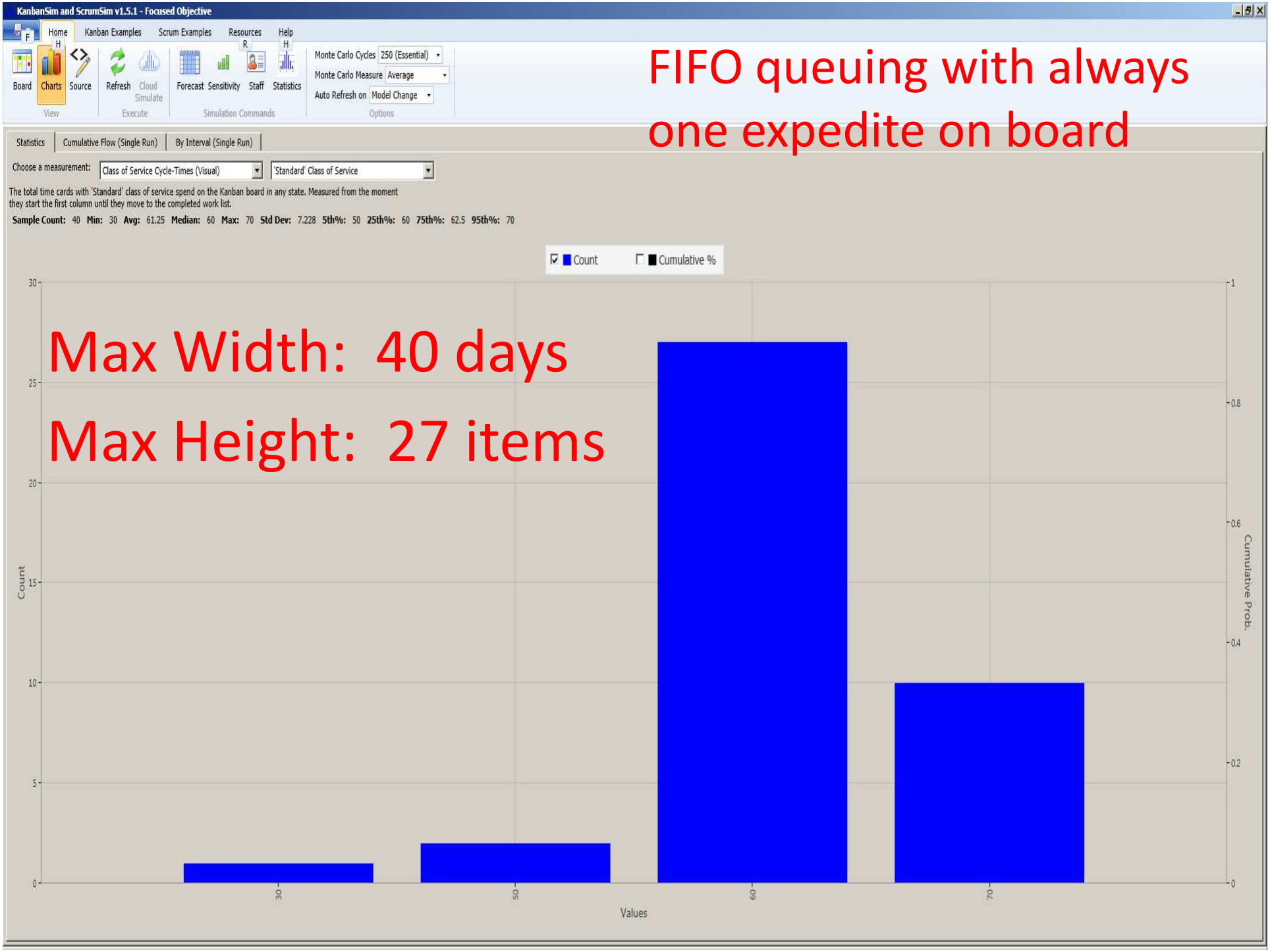
Random Queuing (no expedites)



Random Queuing always one expedite

Let’s look at these in reverse order as that is how most teams start out


KanbanSim and ScrumSim v1.5.1 - Focused Objective
Home
Kanban Examples
Scrum Examples
Resources
Help
Board
Charts
Source
Refresh
Cloud Simulate
Forecast
Sensitivity
Staff
Statistics
View
Execute
Simulation Commands
Monte Carlo Cycles 250 (Essential)
Monte Carlo Measure 95th Percentile
Auto Refresh on Model Change
Options
Random queuing with always one expedite on board
Statistics
Cumulative Flow (Single Run)
By Interval (Single Run)
Choose a measurement:
Class of Service Cycle-Times (Visual)
'Standard' Class of Service
The total time cards with 'Standard' class of service spend on the Kanban board in any state. Measured from the moment they start the first column until they move to the completed work list.
Sample Count: 33 Min: 40 Avg: 72.424 Median: 60 Max: 170 Std Dev: 31.027 5th%: 40 25th%: 50 75th%: 90 95th%: 126
Count
Cumulative %
Max Width: 130 days
Max Height: 7 items
Count
Cumulative Prob.
Values
0
1
2
3
4
5
6
7
0
0.2
0.4
0.6
0.8
1
40
50
60
70
80
90
100
110
150
170



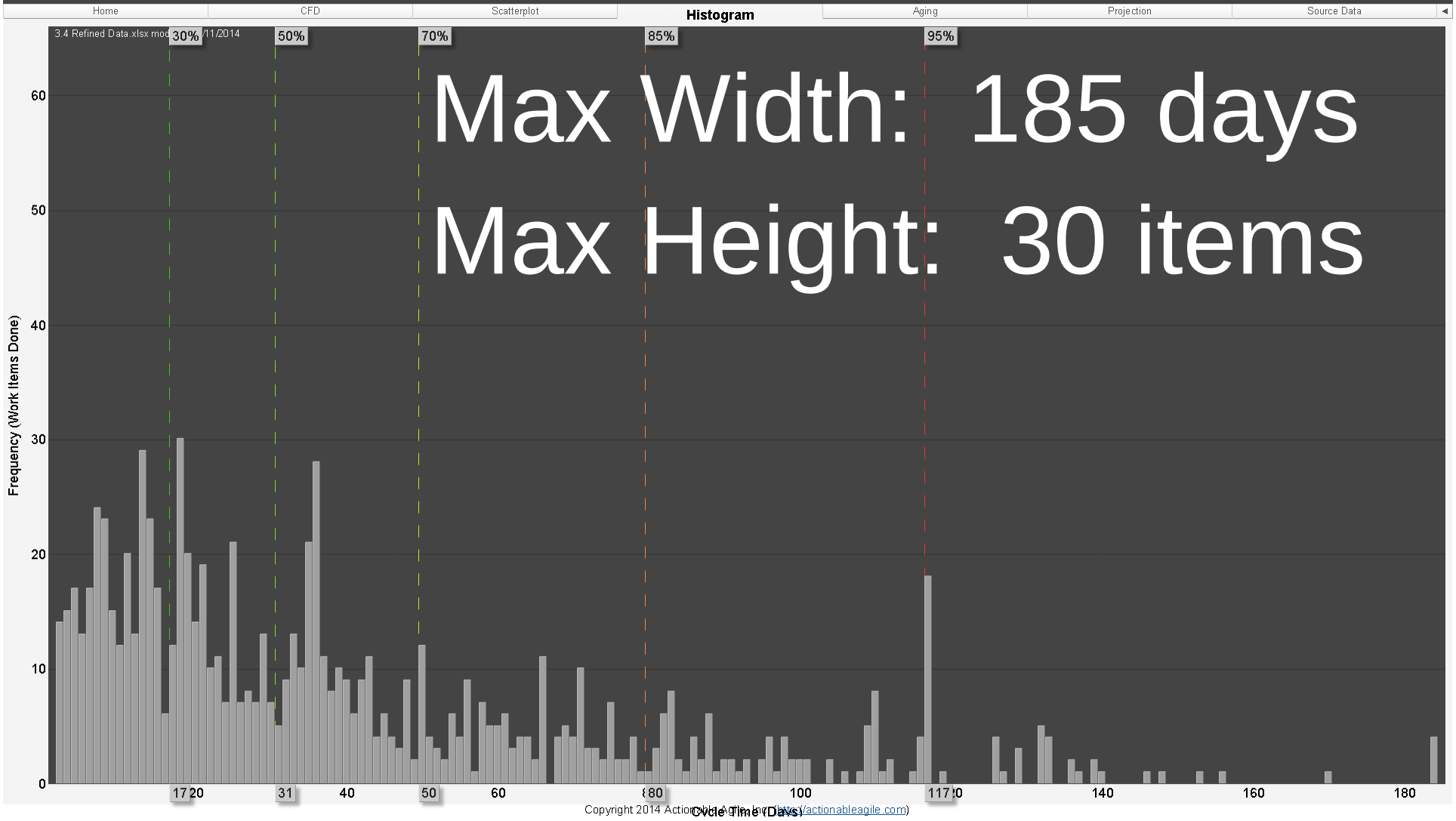
FIFO queuing with always one expedite on board


KanbanSim and ScrumSim v1.5.1 - Focused Objective
Home
Kanban Examples
Scrum Examples
Resources
Help
Board
Charts
Source
Refresh
Cloud Simulate
Forecast
Sensitivity
Staff
Statistics
Monte Carlo Cycles 250 (Essential)
Monte Carlo Measure 95th Percentile
Auto Refresh on Model Change
View
Execute
Simulation Commands
Options
FIFO Queuing (no expedites)
Statistics
Cumulative Flow (Single Run)
By Interval (Single Run)
Choose a measurement:
Class of Service Cycle-Times (Visual)
'Standard' Class of Service
The total time cards with 'Standard' class of service spend on the Kanban board in any state. Measured from the moment they start the first column until they move to the completed work list.
Sample Count: 50 Min: 30 Avg: 49.2 Median: 50 Max: 50 Std Dev: 3.405 5th%: 44.5 25th%: 50 75th%: 50 95th%: 50
Count
Cumulative %
Max Width: 20 days
Max Height: 47 items
Count
Cumulative Prob.
Values
30
40
50

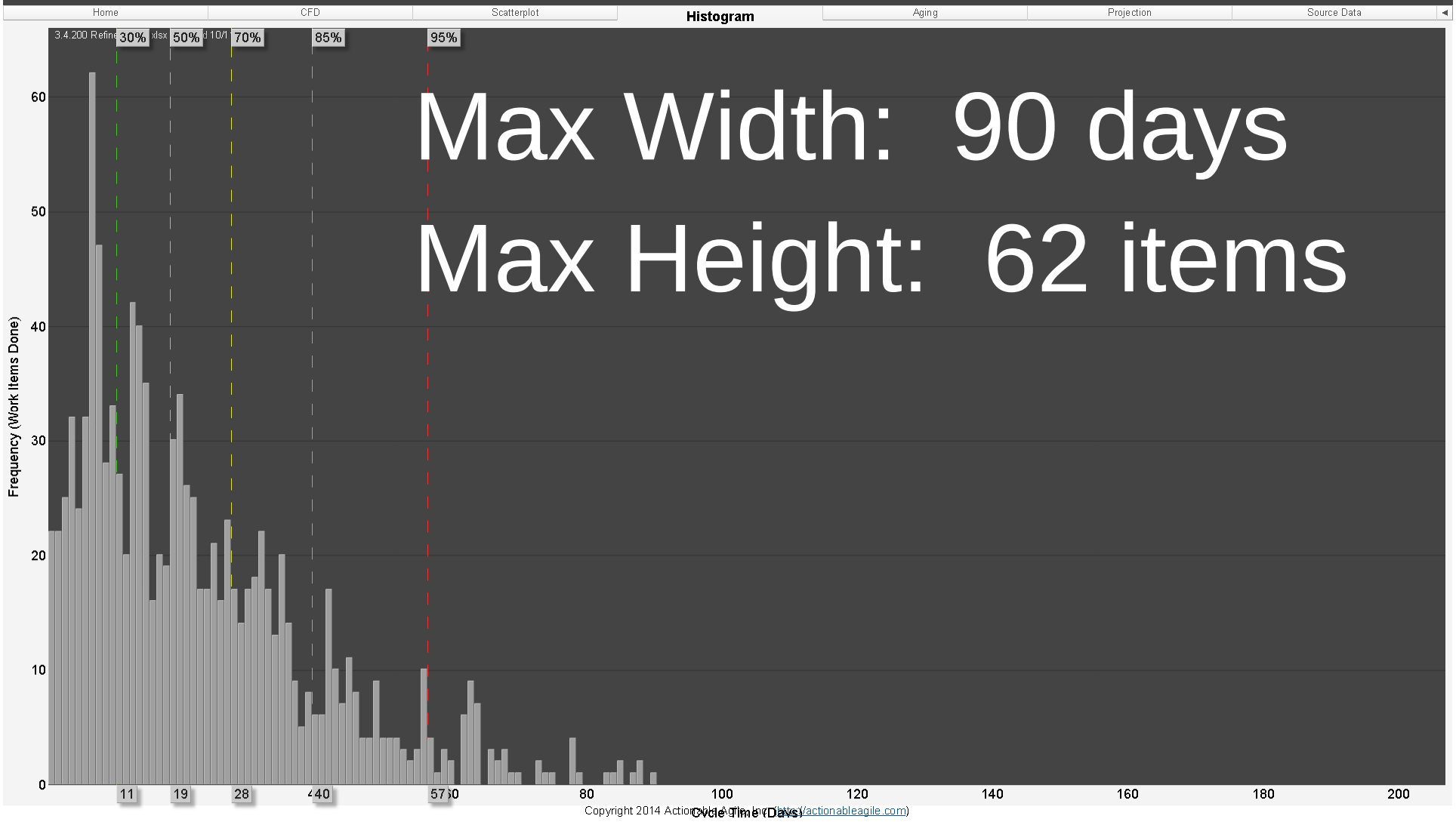
Why Size Doesn’t Matter
Reason #3 :
Poor Pull Policies

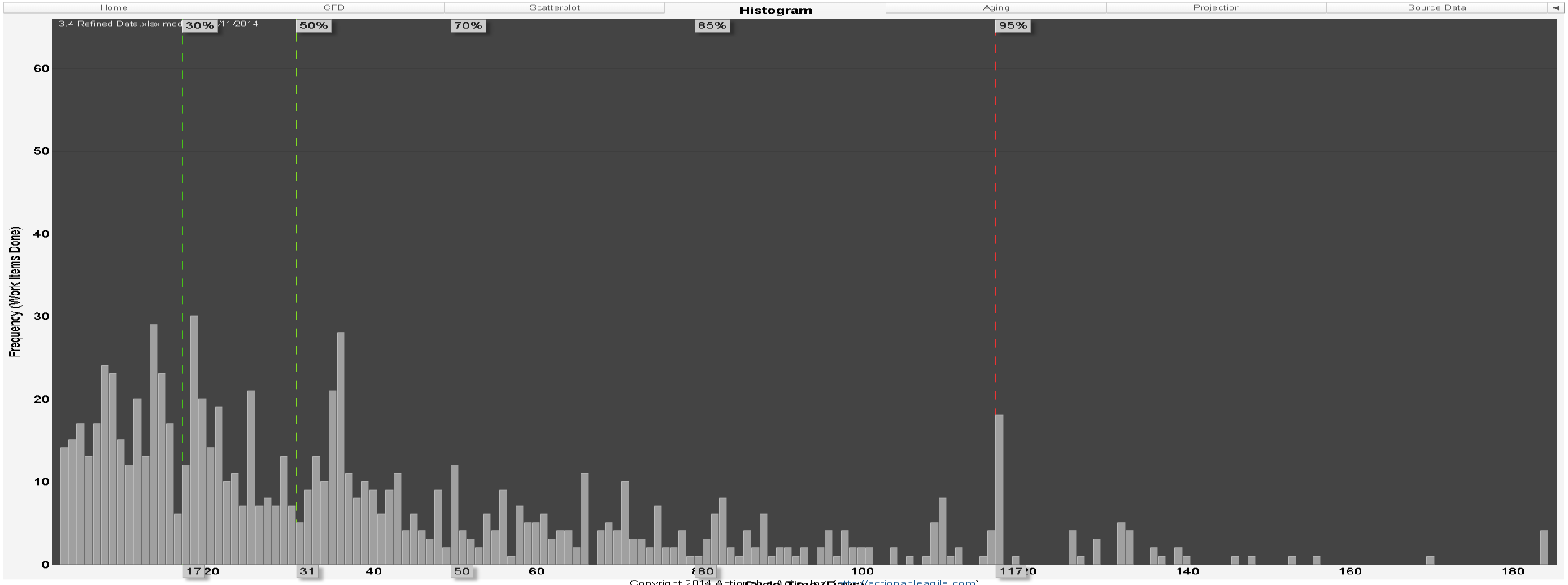
Real World Example

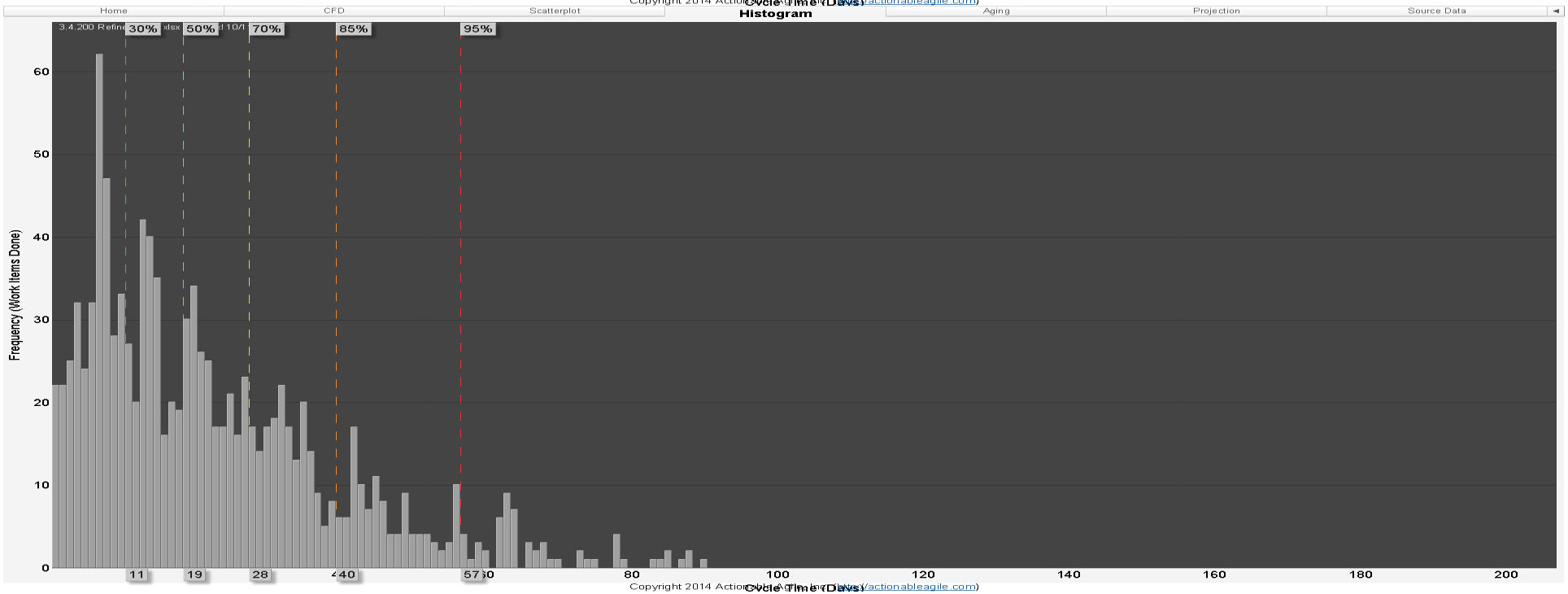
No WIP Limits, No Pull Policies, Doing Estimation



Limiting WIP, Pull Policies, No Estimation




Home
CFD
Scatterplot
Histogram
Aging
Projection
Source Data
3.4 Refined Data.xlsx mod
30%
50%
70%
85%
95%
Frequency (Work Items Done)
0
10
20
30
40
50
60
17
20
31
40
50
60
80
100
117
140
160
180
Cycle Time (Days)
Copyright 2014 Actionable Agile (www.actionableagile.com)


Home
CFD
Scatterplot
Histogram
Aging
Projection
Source Data
30%
50%
70%
85%
95%
Frequency (Work Items Done)
0
10
20
30
40
50
60
11
19
28
40
57
80
100
120
140
160
180
200
Cycle Time (Days)
Copyright 2014 Actionable Agile (www.actionableagile.com)

To Sum Up

The the biggest reasons why traditional sizing doesn’t matter:

- Measuring the wrong thing
- Too much WIP
- Poor Pull Policies

These items have a much bigger impact on elapsed time than estimation of size.

The good news is that these two things are directly under our control.

Take Control of the stuff you can control.

Go back to your team and

Measure the right thing,

Control WIP,

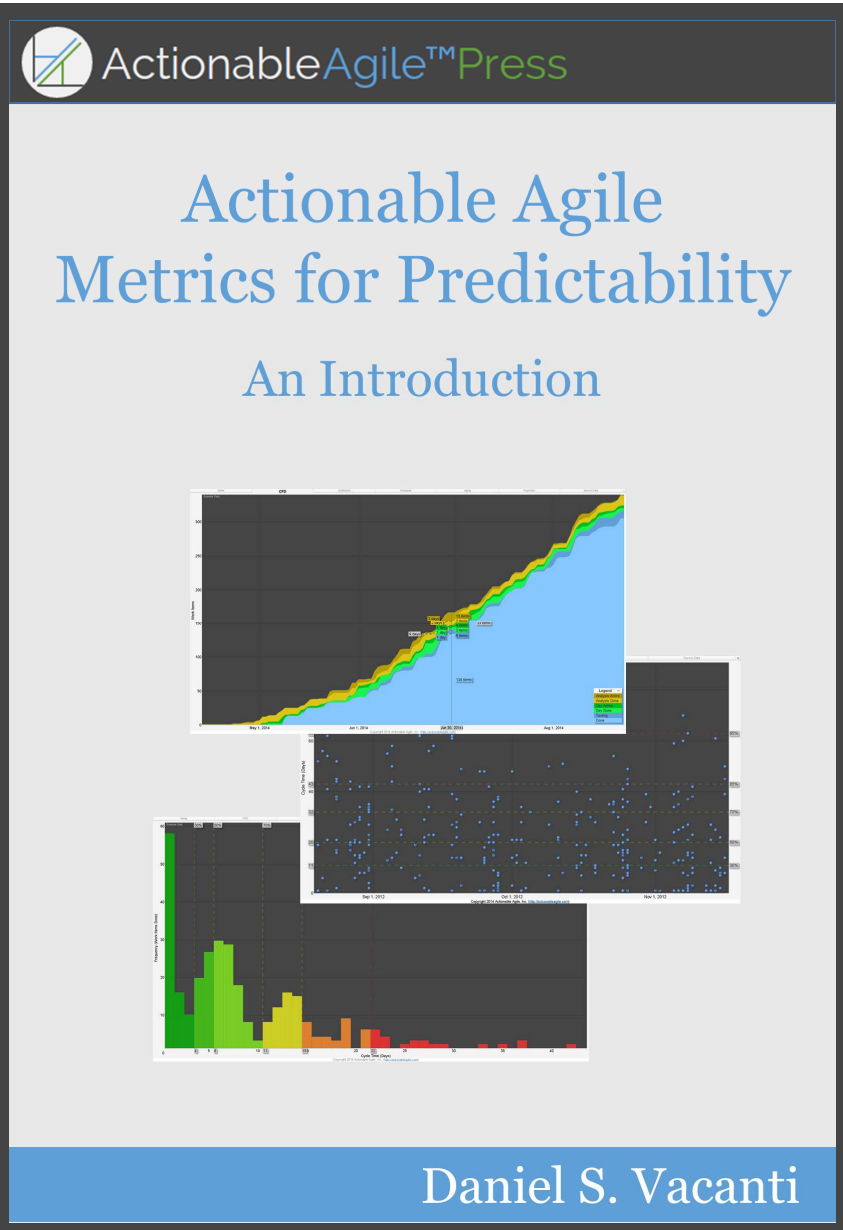
Implement Sane Pull Policies,

and…

Stop Estimating!

For next time…

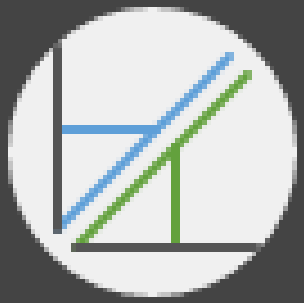
What does a 19th century Yorkshire cotton industrialist have to do with the Manhattan Project?



"Actionable Agile Metrics for Predictability"

https://leanpub.com/actionableagilemetrics

All charts created by:



ActionableAgile

Take Control

THANK-YOU!

Daniel S. Vacanti
http://www.actionableagile.com
daniel@actionableagile.com
@danvacanti