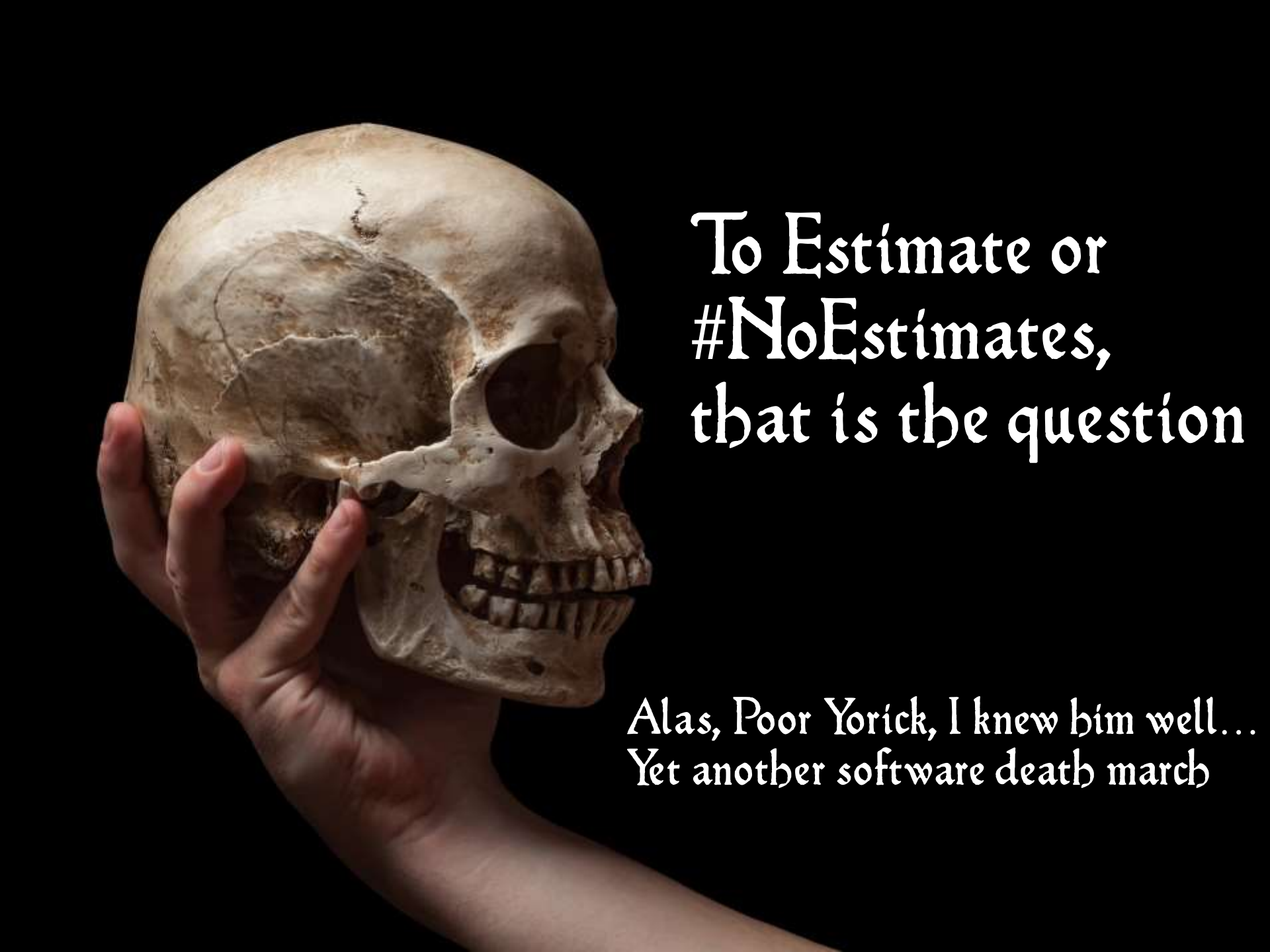

To Estimate or
#NoEstimates,
that is the question
Alas, Poor Yorick, I knew him well…
Yet another software death march

To Estimate or #NoEstimates, That is the Question

Sean Dunn

Enterprise Agile Coach

Todd Little

VP Product Development

IHS

Chris Verhoef

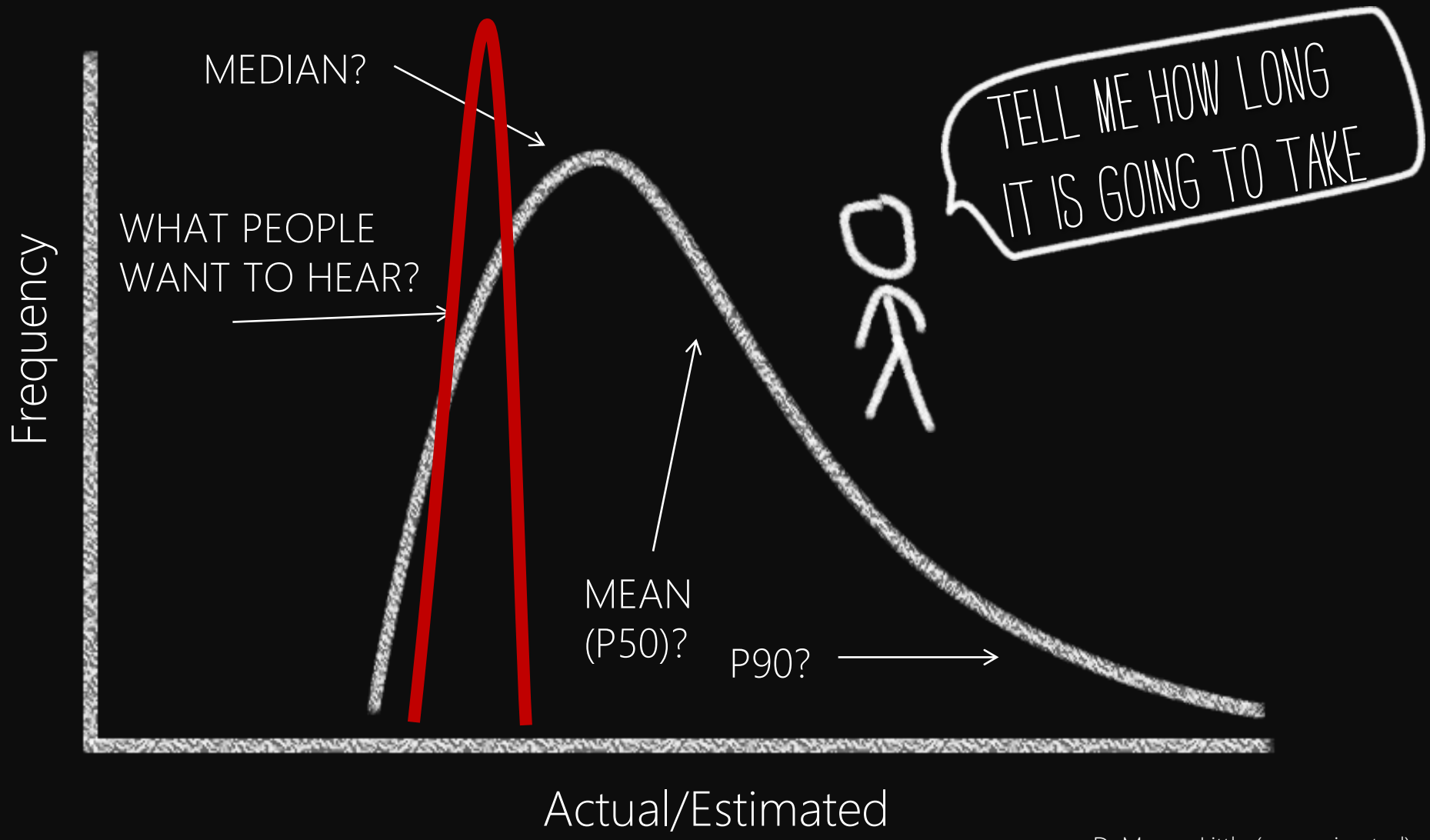
Vrie University, Amsterdam

How many of you consider it to be a major part of your job to either create estimates or deliver against someone's estimates?

“It is difficult to get a man to understand something when his salary depends upon his not understanding it.”

Upton Sinclair

Probability Distribution Estimate/Actual



DeMarco, Little (approximated)

“Without risk, information would literally have no value to decision making.”

Douglas Hubbard,
How to Measure Anything

What do we Estimate?

What do we Estimate?

- Release – duration, effort, cost
- Feature/Epic – story points, T-shirt sizes
- Stories – story points
- Tasks – hours

Why do we Estimate?

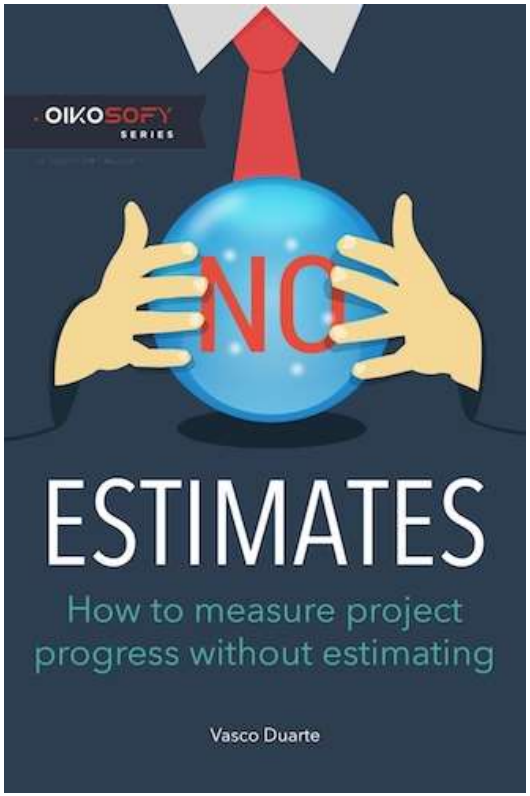
Why do we Estimate?

- Decisions we need to make
  - Do we start at all?
  - What should we work on next?
  - Should we stop and work on something else?
  - Should we swarm?
  - Should we get help?
  - Should we split?
  - Should we re-evaluate our technical approach?

What is #NoEstimates

"#NoEstimates is a hashtag for the topic of exploring alternatives to estimates [of time, effort, cost] for making decisions in software development. That is, ways to make decisions with 'No Estimates'."

Woody Zuill



One Approach Suggested by #NoEstimates

- Discontinue estimating story points
- Instead simply count number of completed stories per iteration (throughput)

We collected and analyzed some real data

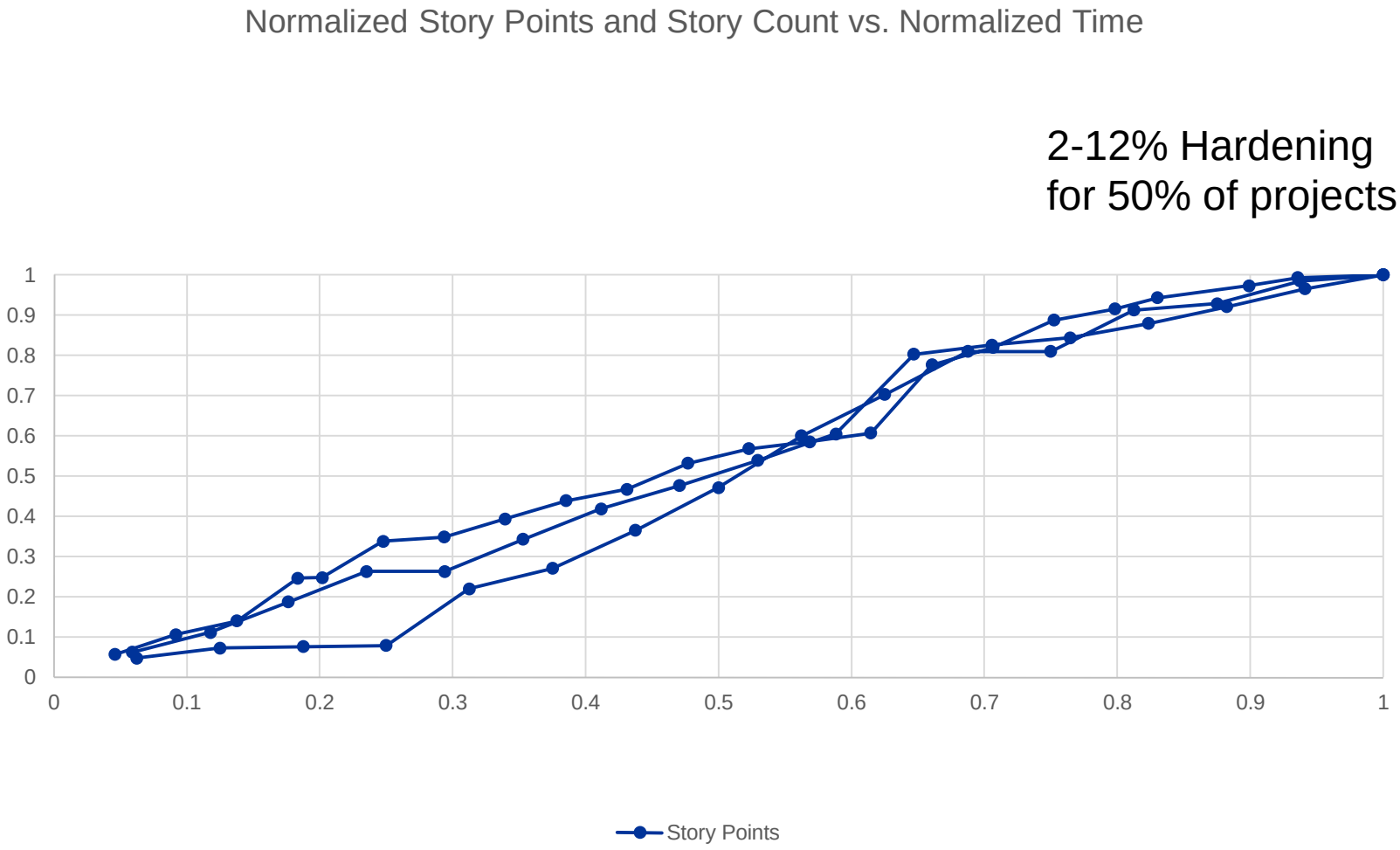
About the Data

- Data collected anonymously by Vasco Duarte
- 55 Projects from 9 companies
- 37 projects came from one organization
- …..but the analysis of the data is consistent
- https://bitly.com/NoEstimatesProjectsDB

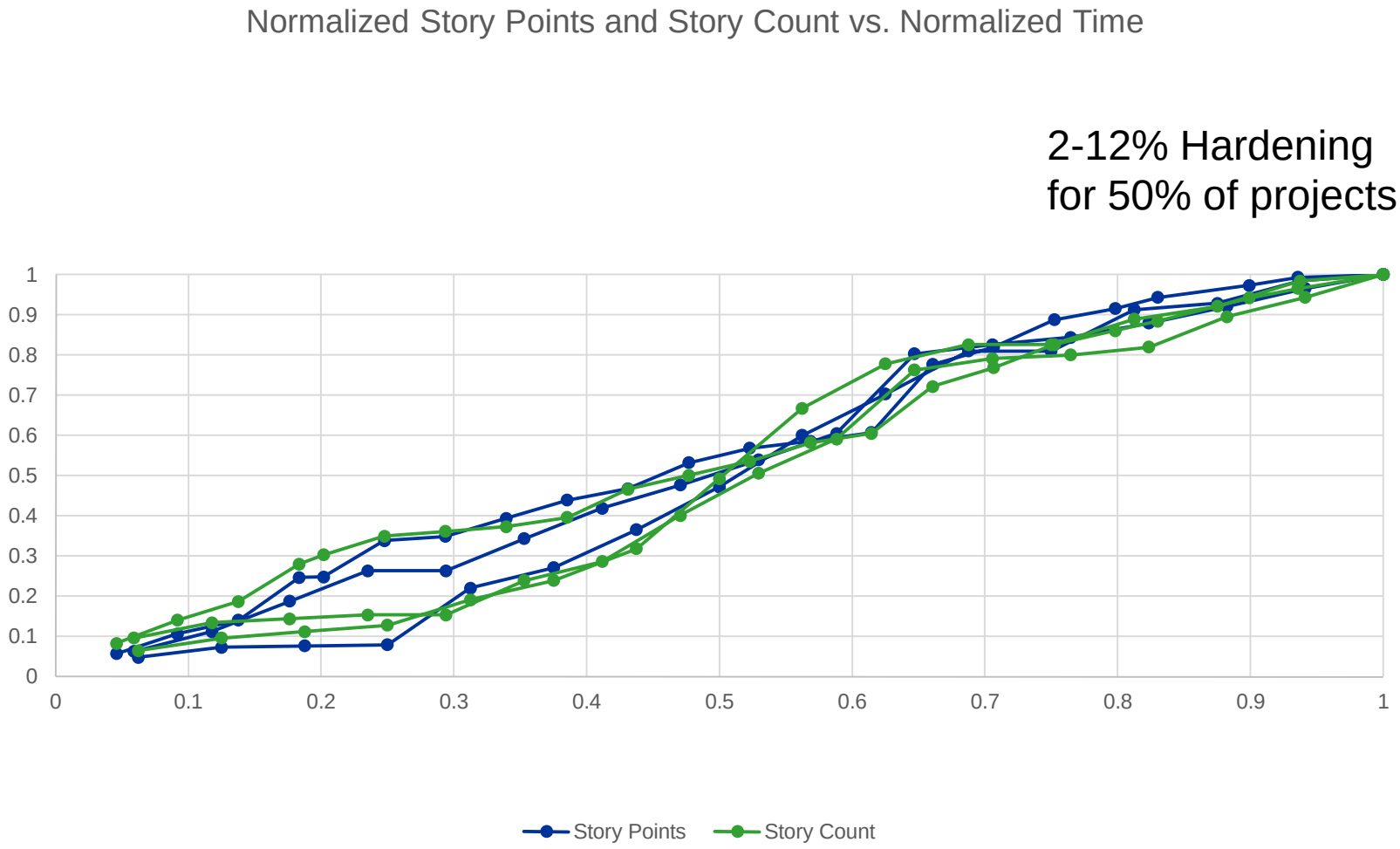
Definitions

	Instantaneous	Average
Velocity	Story points delivered in an iteration	Average story points delivered over all iterations
Throughput	Number of stories delivered in an iteration	Average number of stories delivered over all iterations

What does the Data show?



What does the Data show?



P90/P10 Ratio

P10=10% confidence

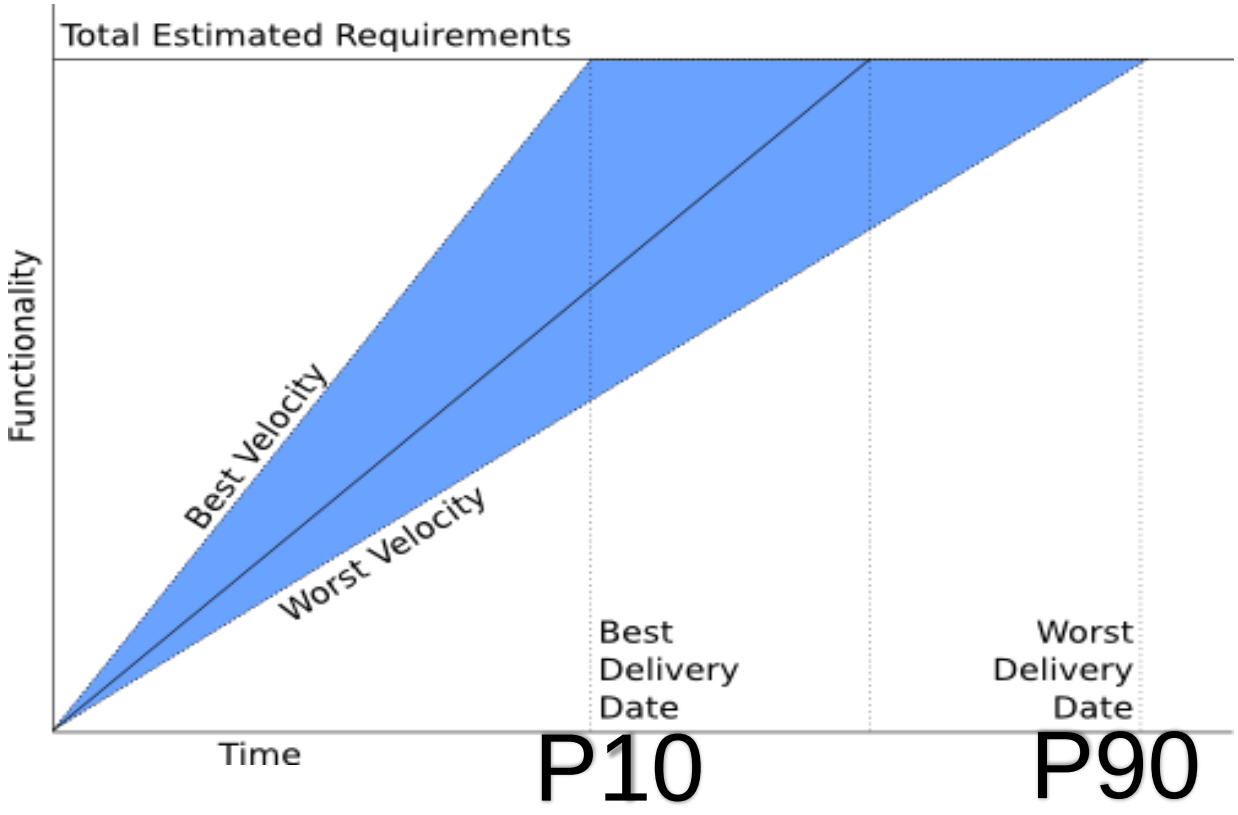
P90=90% confidence

- Example

P10 = 6 months

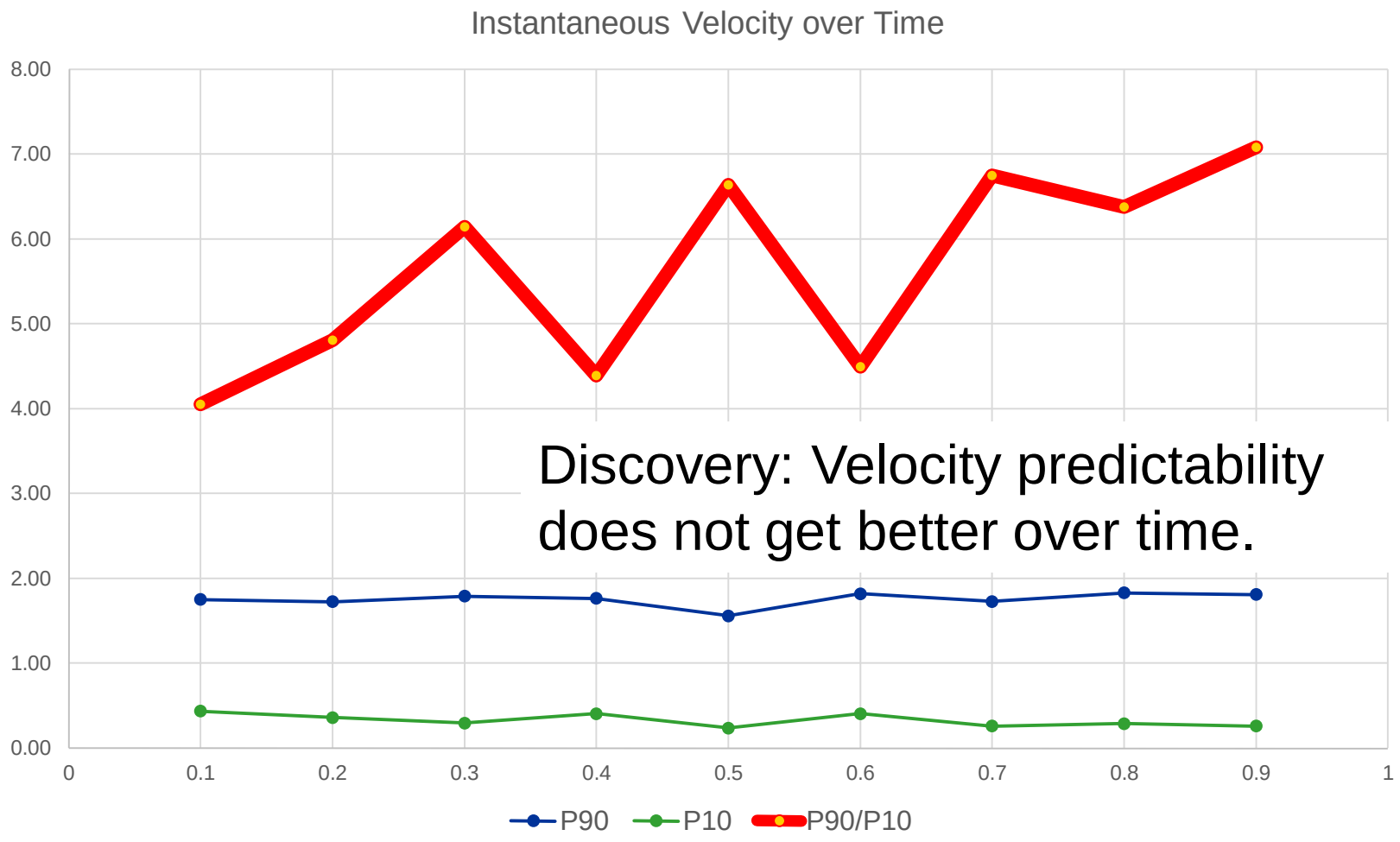
P90 =12 months

P90/P10 = 2



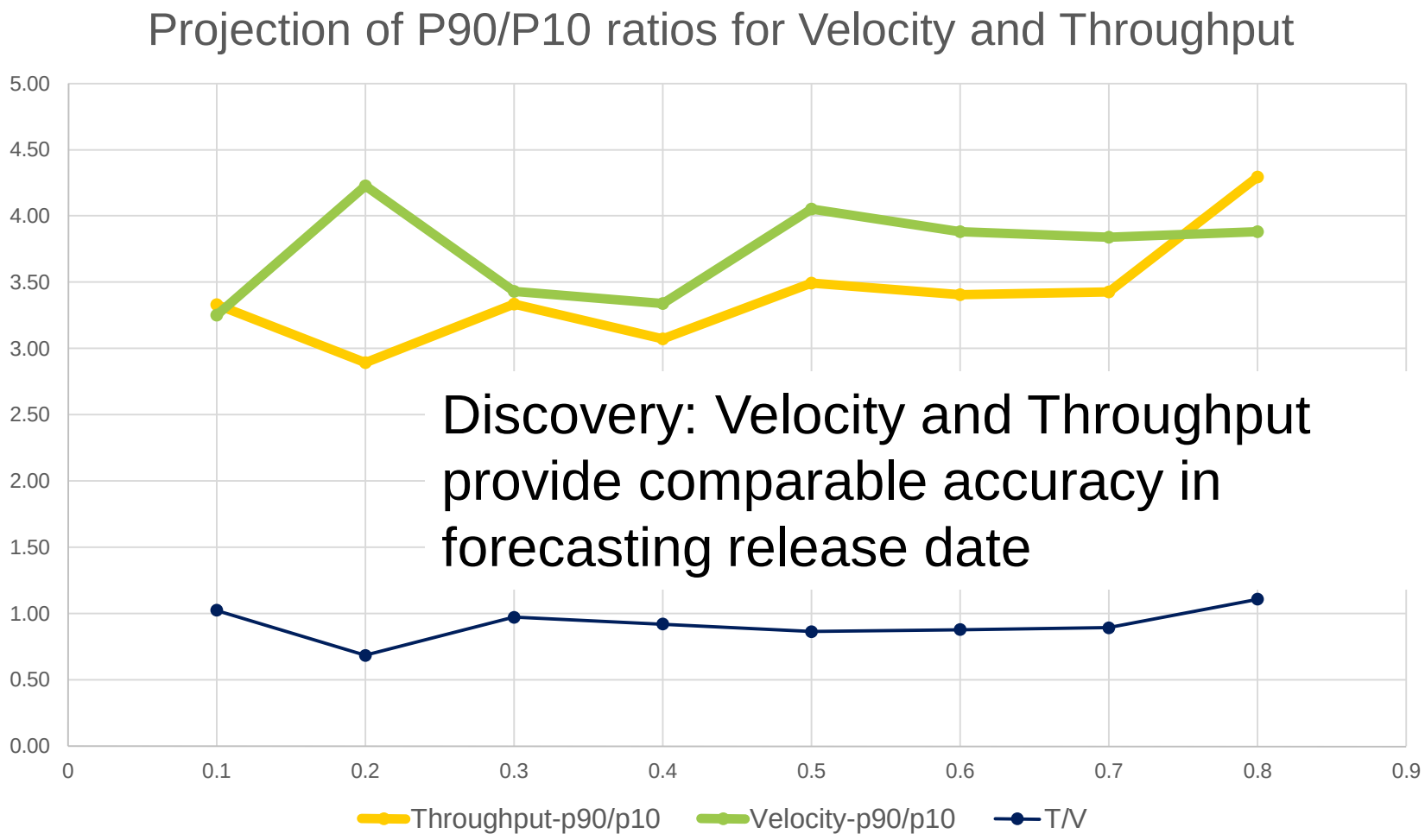
Question:
Do we get better at estimating the further we go in a project?
(or: does our velocity become more stable over time?)

What do we see in Velocity?



Question:
If we just use throughput (story count) to forecast completion, is it any more/less accurate than using velocity (story points)?

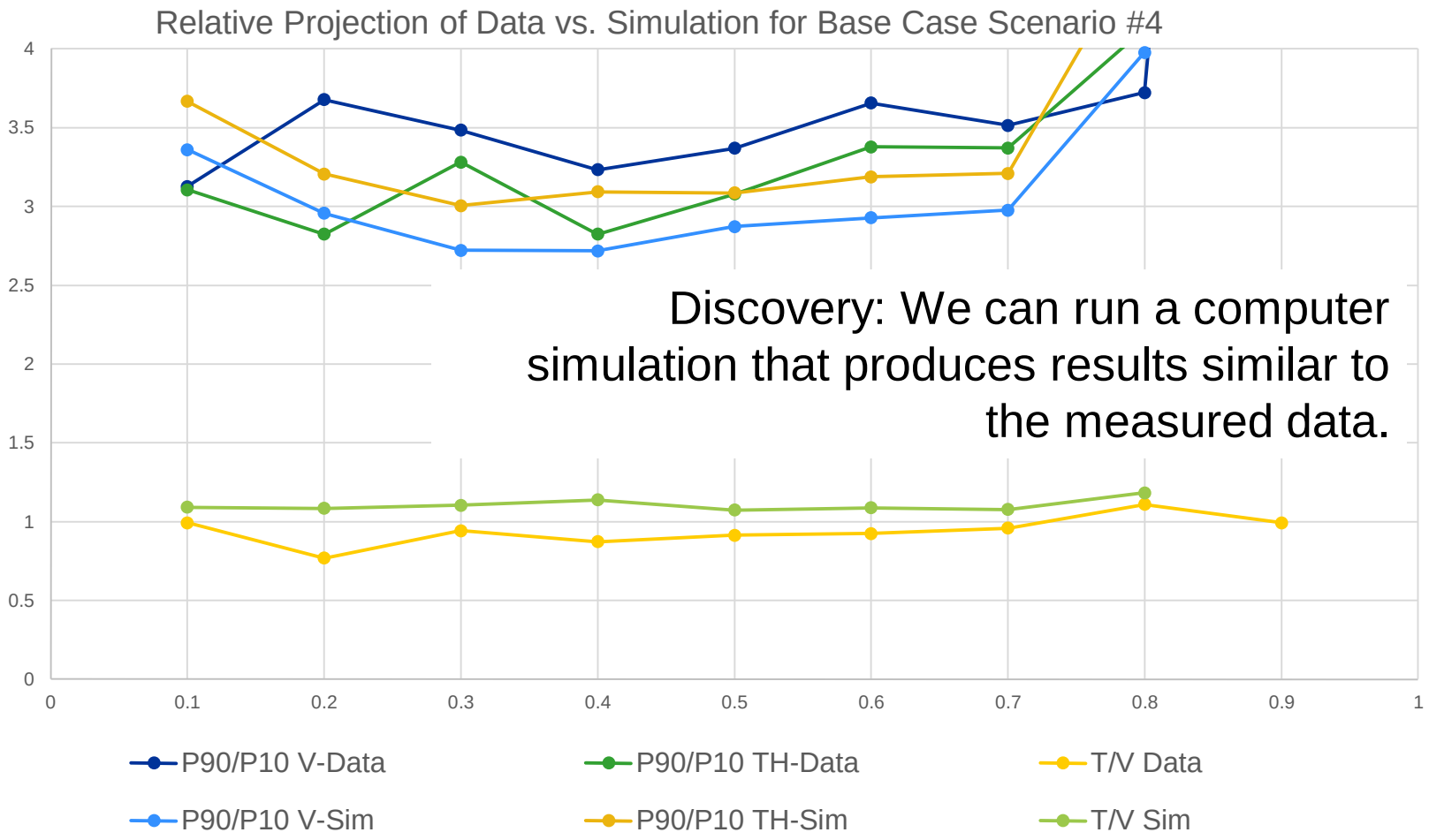
What do we see in Projections?



How about a Simulation

- Simulated 1000 projects with 50 stories per project
  - Story Point Distribution
  - Estimation Accuracy
  - Bucketing Approach
  - Hardening Effort

How about a Simulation



Discovery: We can run a computer simulation that produces results similar to the measured data.

Simulation Results

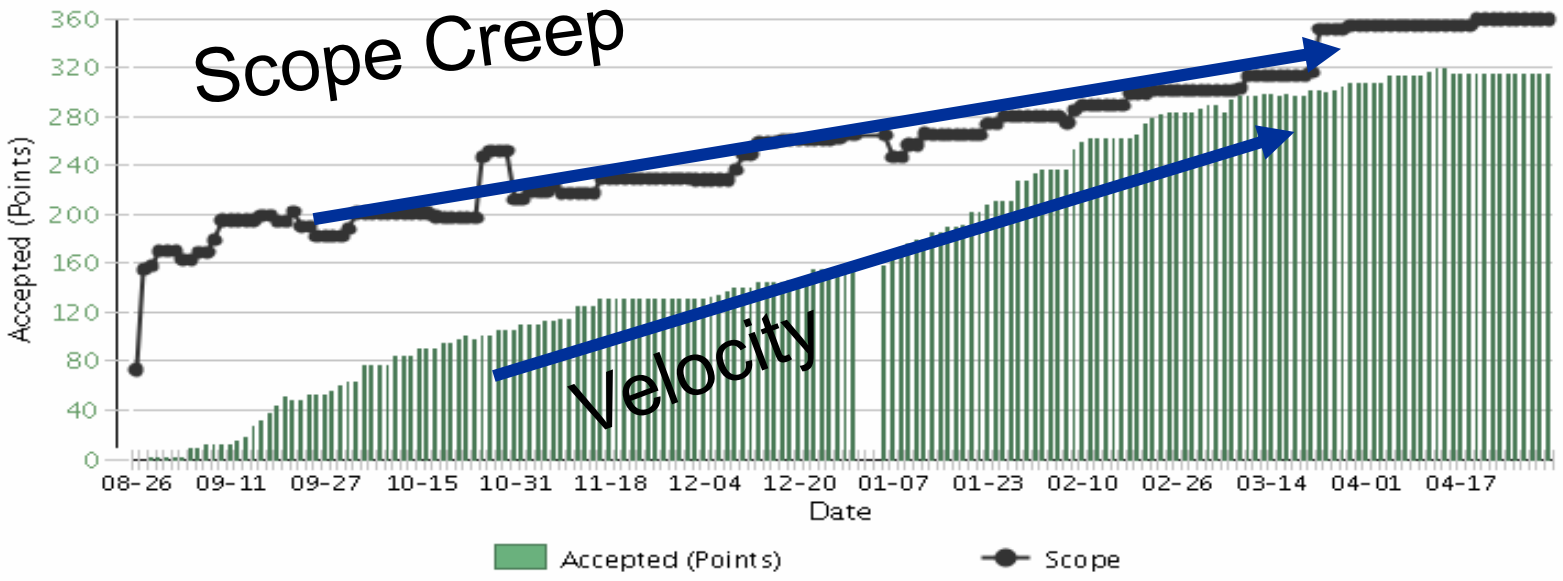
- No fundamental difference between velocity and throughput projections.
- Velocity showed improvement over throughput when:
  - Story point distribution was very large
  - Estimation Accuracy was very good
- Using buckets, such as Fibonacci, power of 2, or even power of 4 did not significantly impact the simulation.

What do the simulations tell us

- Estimating Mixed nuts…..

What does this tell us about Estimation?

Decisions to steer towards the release	**Velocity and Throughput are equally good AND equally bad predictors, but better than nothing.**

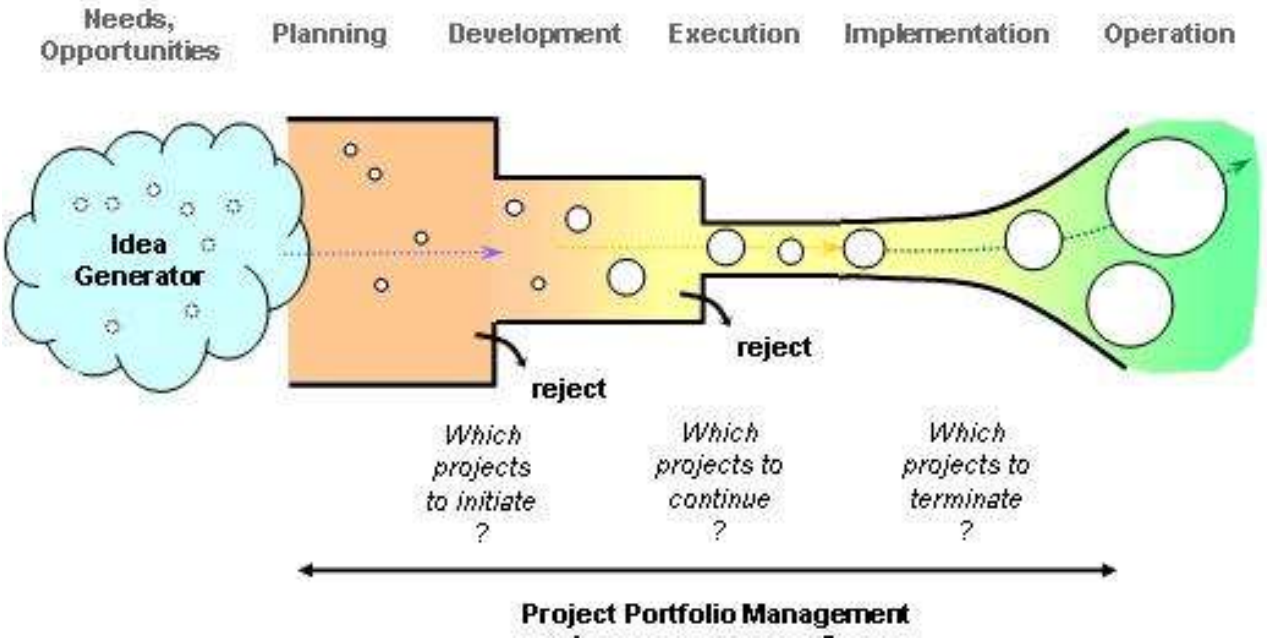


What does this tell us about Estimation?

Decisions to help with managing iterations	Do task estimates add value? What decisions are you making based on what you assume?

What does this tell us about Estimation?

Decisions at project sanction	**Some level of macro-estimation of costs and benefits is likely necessary for business decisions.** **In general it is waste to spend more time on cost estimation than on benefits.**



Comparison to other Research

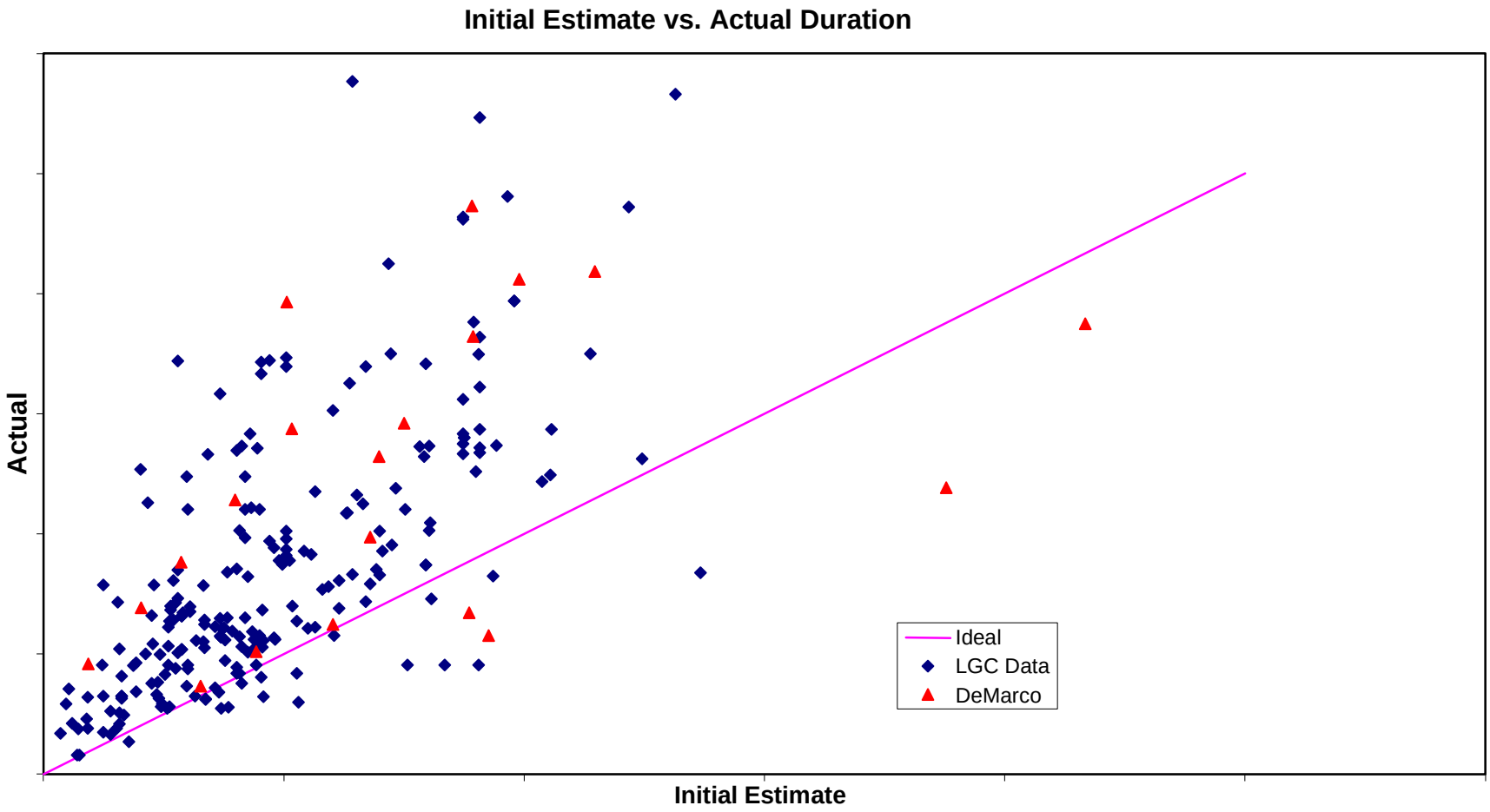
feature

project estimation

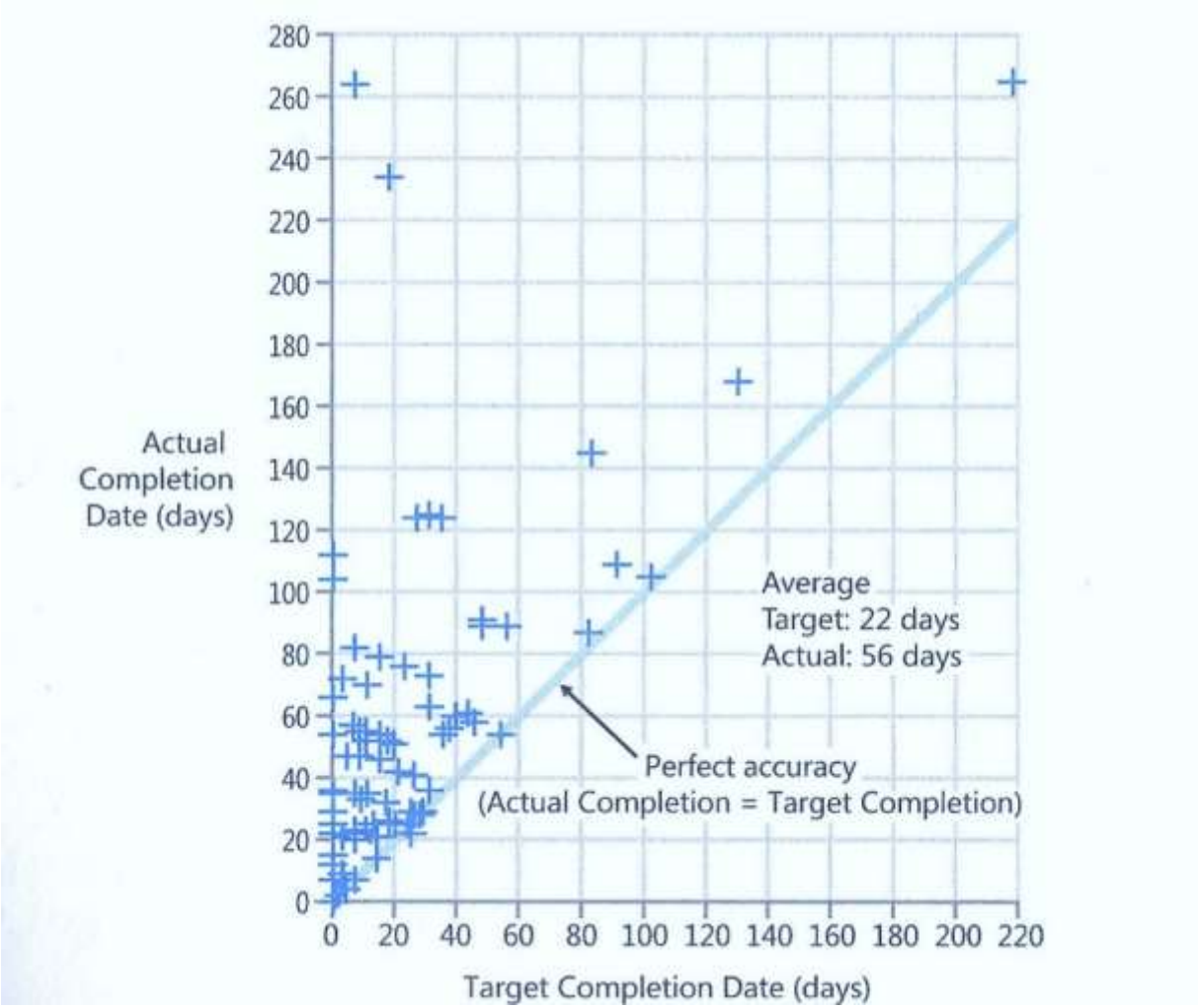
Schedule Estimation and Uncertainty Surrounding the Cone of Uncertainty

Todd Little, *Landmark Graphics*

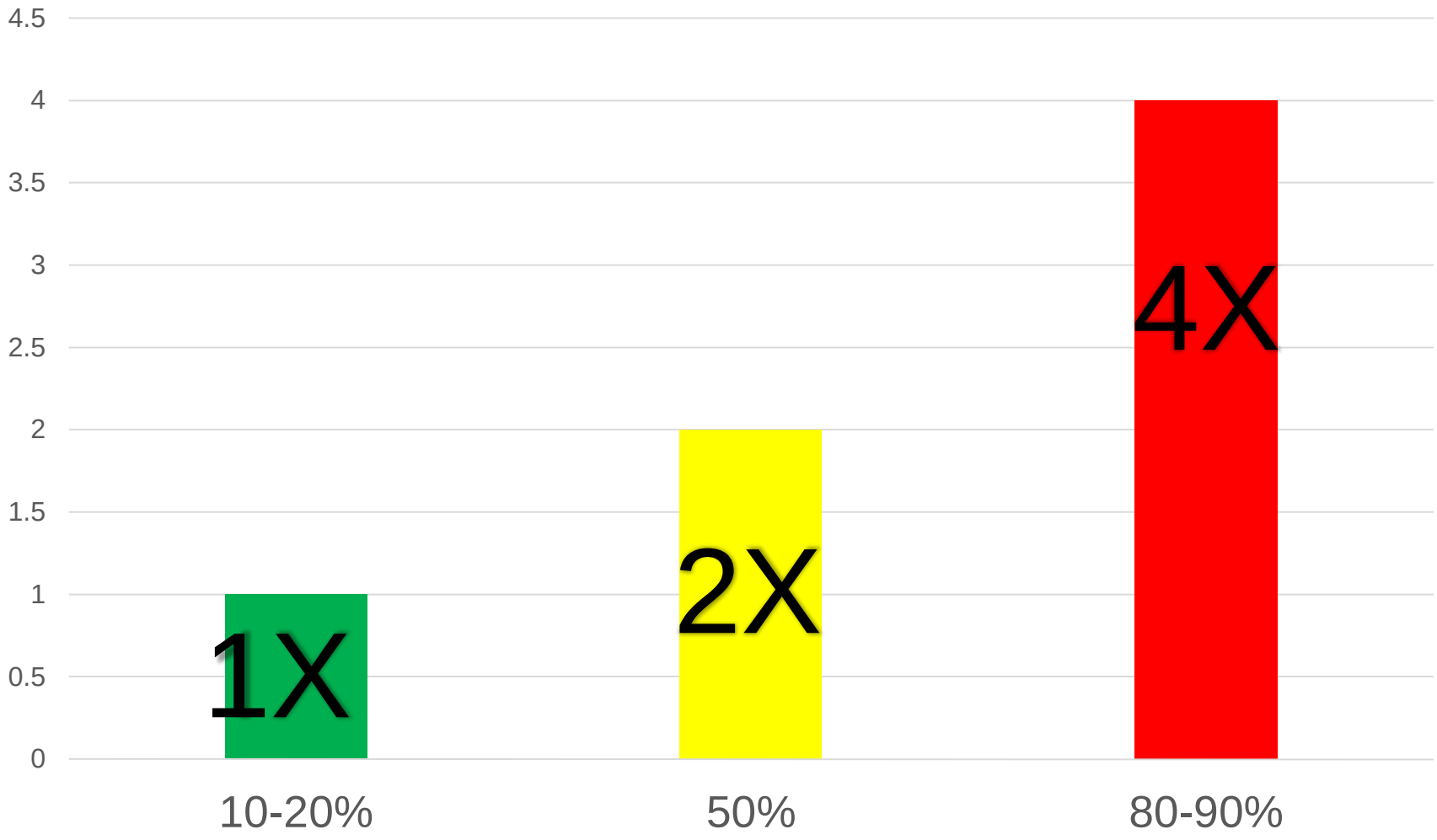
Accuracy of Initial Estimate



Data From Steve McConnell



Actual vs. Original Estimate

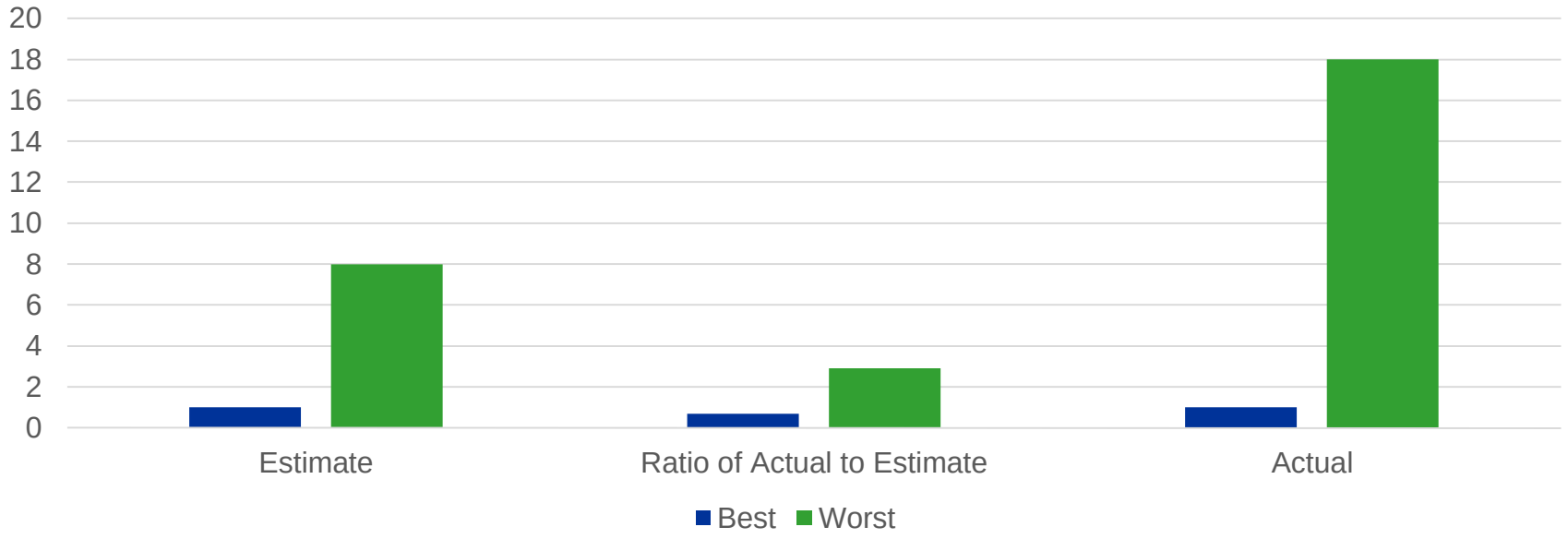


Jørgensen 2013

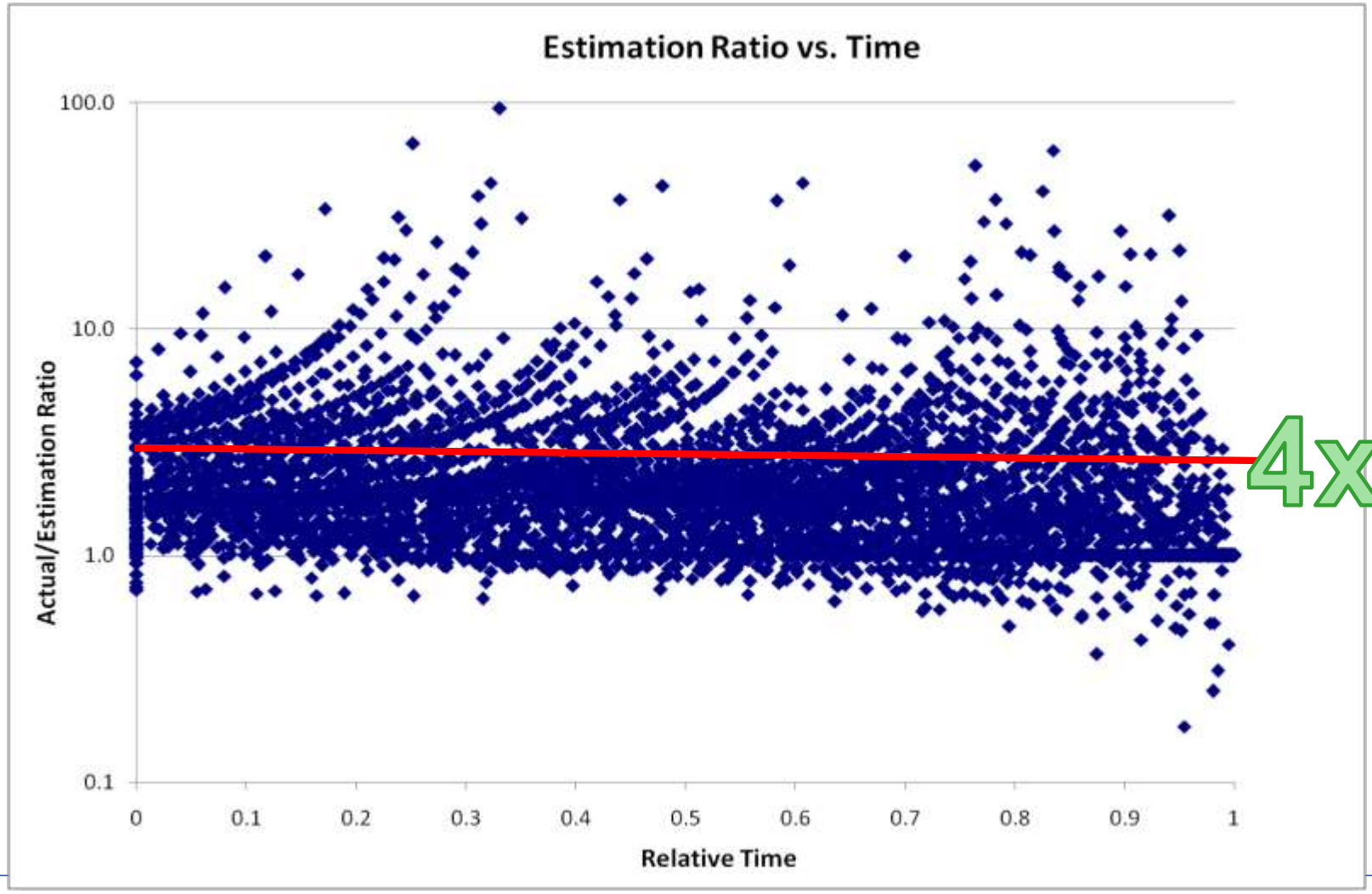
- Put software development project for bid on online marketplace vWorker.com
- Received 16 bids.
- Reduced down to 6 bids from vendors that had high (9.5) client satisfaction.
- All 6 bidders went ahead and built the software

Jørgensen 2013

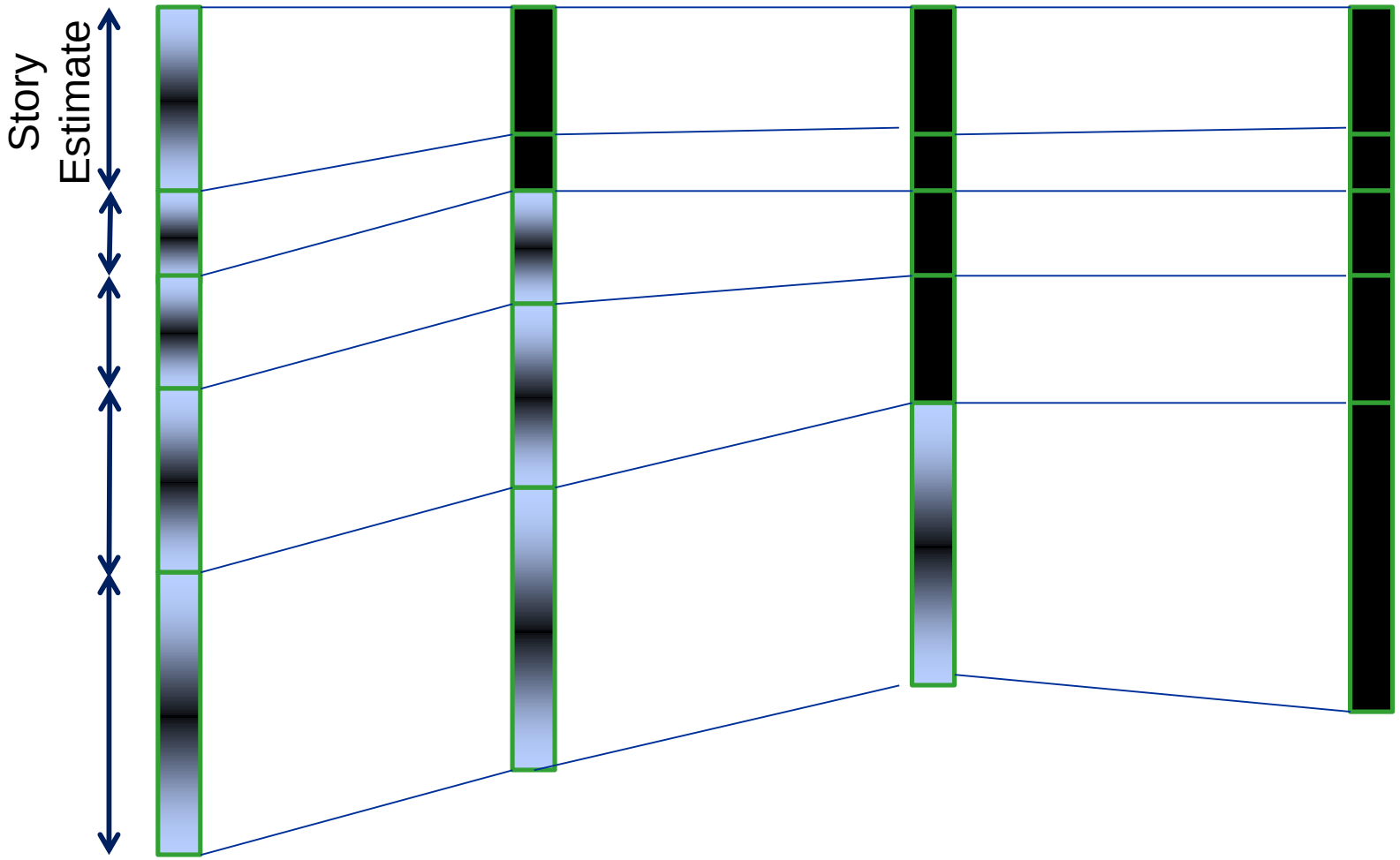
- Highest Estimate 8x the Lowest
- Actual/Estimate Range: 0.7 – 2.9 (4x)
- Actual Performance Range: Worst took 18X the effort of the best



Remaining Uncertainty



Remaining Uncertainty



- How many of you consider it to be a major part of your job to either create estimates or deliver against someone's estimates?

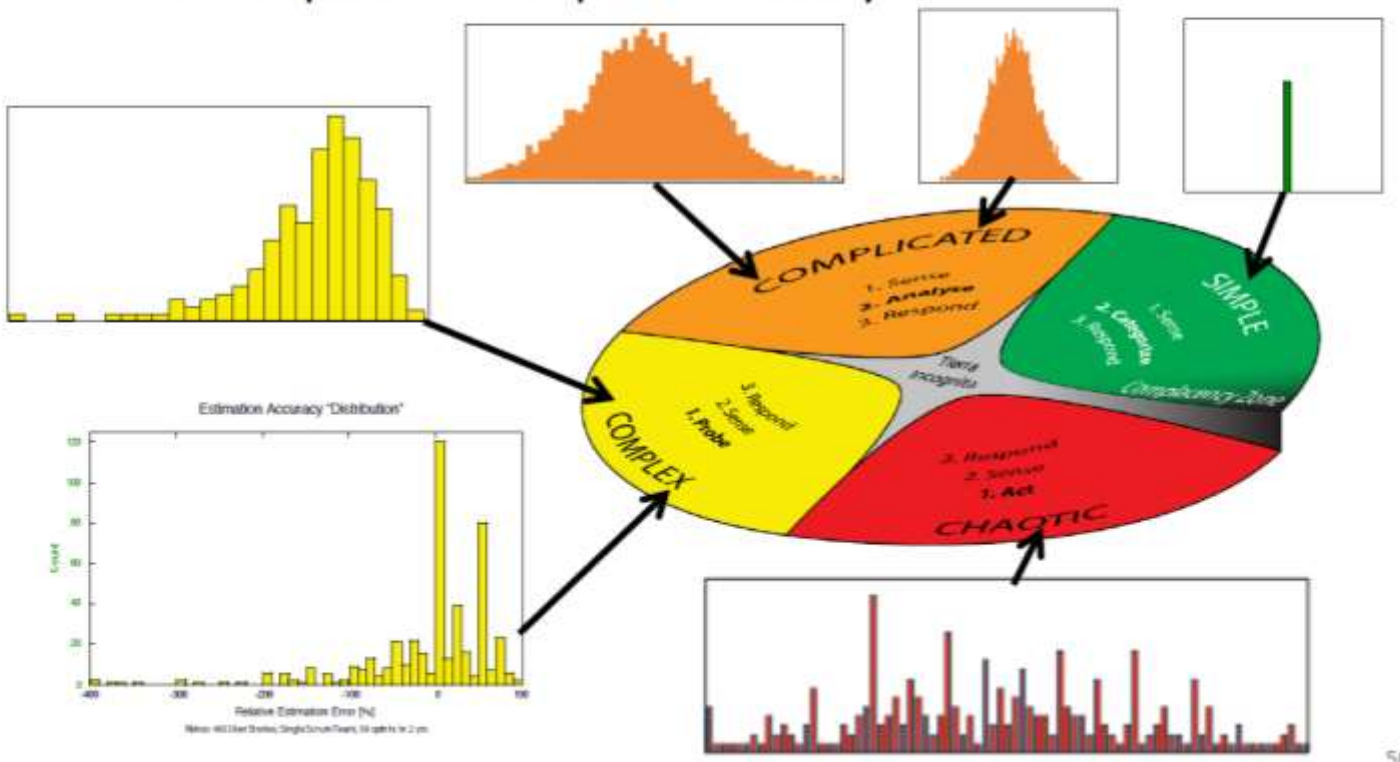
- Now What?

Management Under Uncertainty

Decisions and Projects Measure Unpredictability

Long tails deprive systems of *consistent* predictability

Waterfall requires LOTS predictability



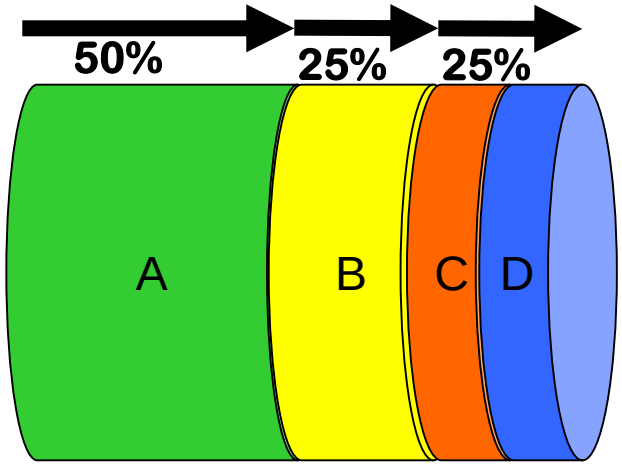
The A/B/C List sets proper expectations

A	MUST be completed in order to ship the product and the schedule will be slipped if necessary to make this commitment. The Product Owner will take the heat for the schedule slippage.
B	Is WISHED to be completed in order to ship the product, but may be dropped without consequence.
C	Is NOT TARGETED to be completed prior to shipping, but might make it if time allows.

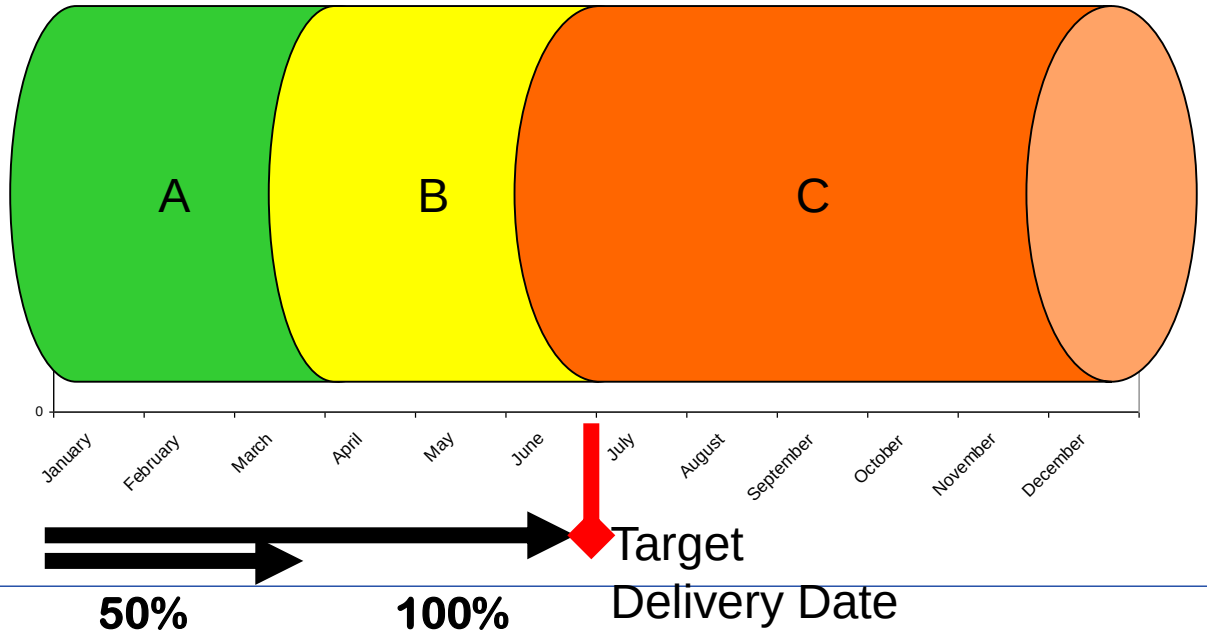
Only “A” features may be committed to customers.

If more than 50% of the planned effort is allocated to “A” items the project is at risk.

A/B/C List

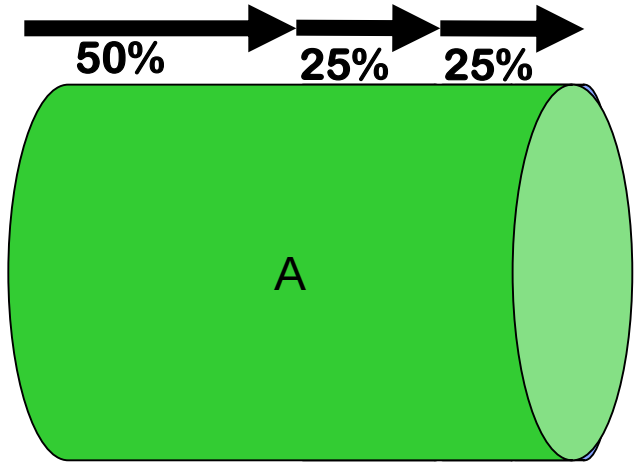


Typical Delivery

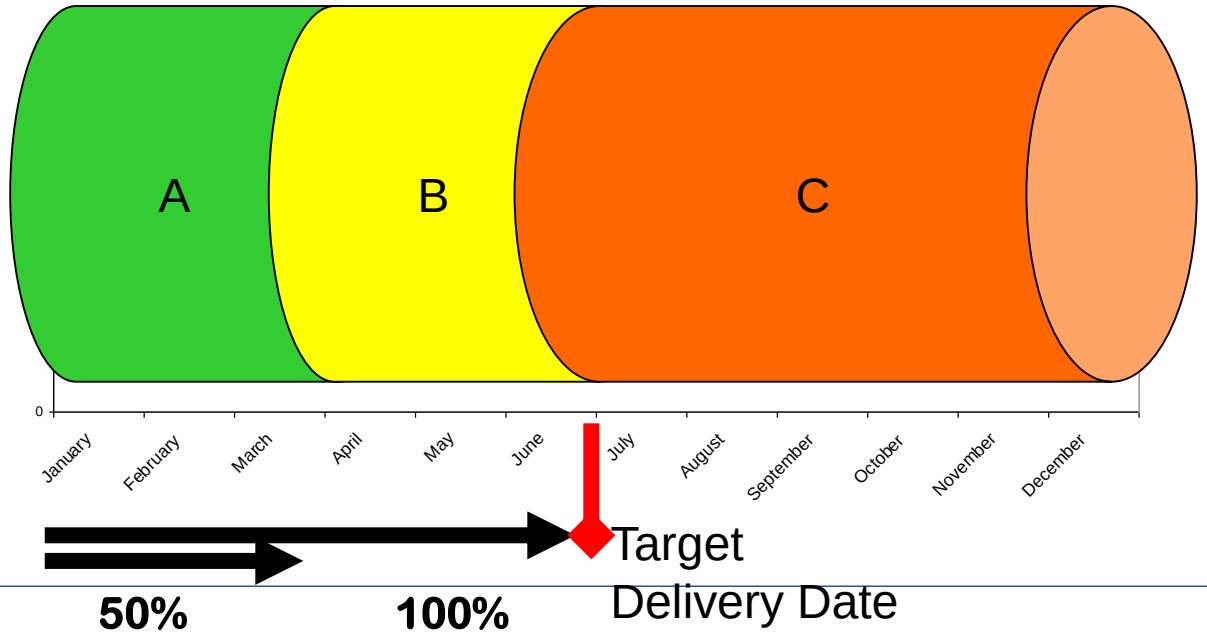


Backlog Plan

A/B/C List



Uncertainty Risk



Backlog Plan

What would Polonius Say?

“Neither an Estimator nor a #NoEstimation bigot be, for estimation oft implies a false sense of both accuracy and certainty, while NO estimates may make suboptimal decisions. To thine own self (and team) be true.”