



# Welcome to the PMI Agile Webinar

- This month's topic: Beyond Scope, Schedule & Cost:  
Rethinking Performance

Thanks to our sponsor



# BEYOND SCOPE, SCHEDULE, AND COST: RETHINKING PERFORMANCE

Jim Highsmith, ThoughtWorks



# “There is no more Normal”

“Without exception, all of my biggest mistakes occurred because I moved too slowly.”

--John Chambers, Cisco CEO

“300 start-ups ... persisting with the initial business plan was the best single predictor of failure.”



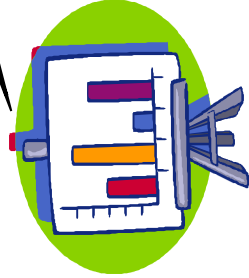
# A Measure of Success



***“I recently asked a colleague [CIO] whether he would prefer to deliver a project somewhat late and over-budget but rich with business benefits or one that is on-time and under-budget but of scant value to the business. He thought it was a tough call, and then went for the on-time scenario. Delivering on-time and within budget is part of his IT department’s performance metrics. Chasing after the elusive business value, over which he thought he had little control anyway, is not.”***

**Cutter Sr. Consultant Helen Puksza**

# Mixed Messages



**Conform to Plan**

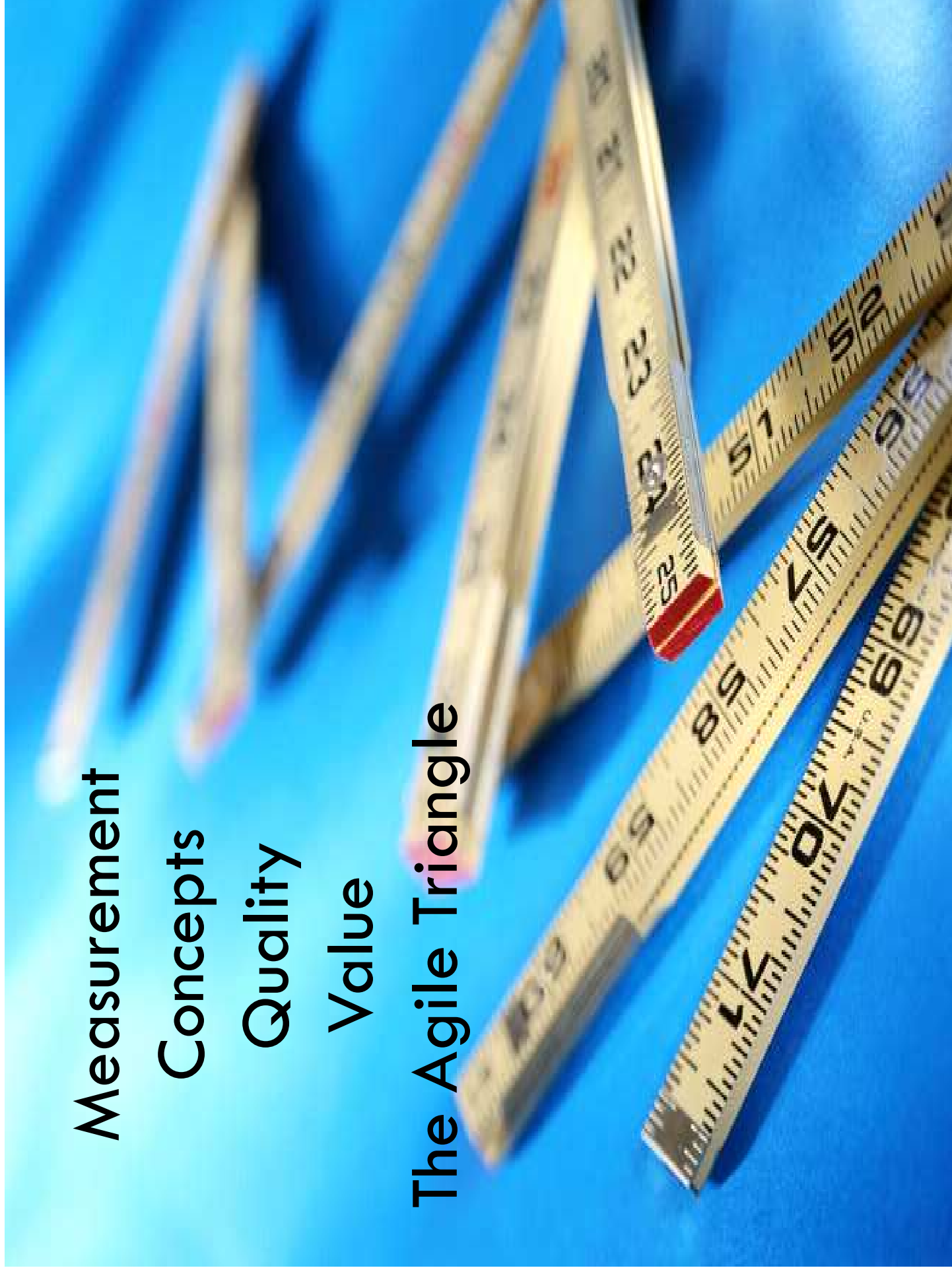


**Be Flexible**



Measurement  
Concepts  
Quality  
Value

The Agile Triangle



# Standish Reports

- Standish Group “Chaos Reports”
  - 1994 — 82% challenged or failures
  - 2001 — 72% challenged or failures
  - 2009 — 68% challenged or failures
- Definition of project “success”
  - Successful: on time, on budget, all specified features;
  - Challenged: completed and operational, but over budget, late, and with fewer features and functions than initially specified;
  - Failed: canceled before completion or never implemented.

**The Standish data are NOT a good indicator of poor software development performance. However, they ARE an indicator of systemic failure of our *planning and measurement processes.***

# Which is Better?

**If higher numbers are improvements, then which of these is better performance?**

**Budget 100 and achieve 100, or  
Budget 120 and achieve 110?**

**Which one would your performance measurement system reward?**



*Aristotle*

**Our problem is not that we aim too high and miss, but that we aim too low and hit.  
--Aristotle**

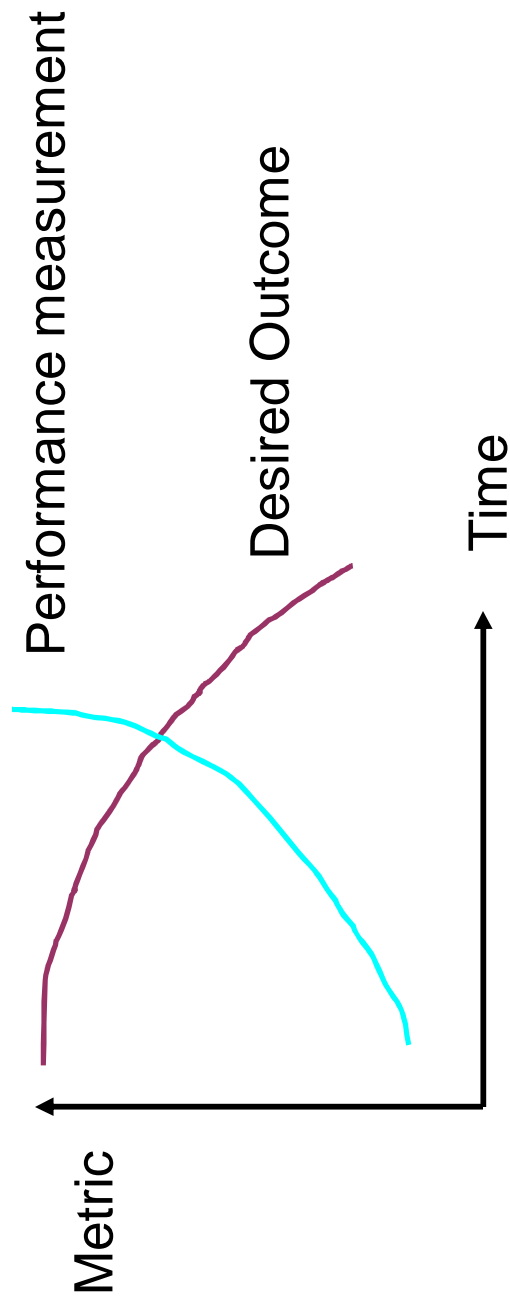
## Austin — How Measurement Systems Become Dysfunctional

Step 1: Measurement system installed.

Step 2: Performance tends to improve while people figure out the system.

Step 3: People, under pressure, focus on measurement goals rather than outcomes. (Always a disconnect between the desired outcome and the measurement. Example: (1) productivity; lines of code.

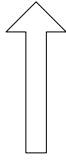
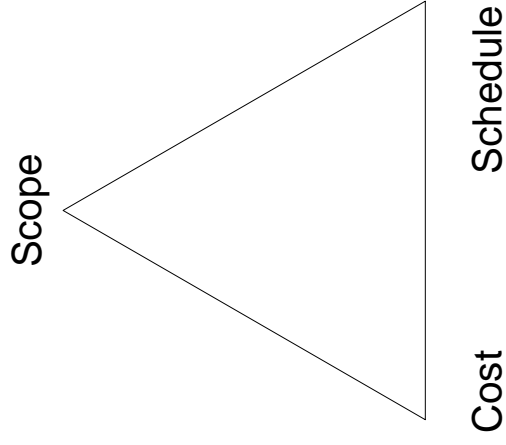
Step 4:



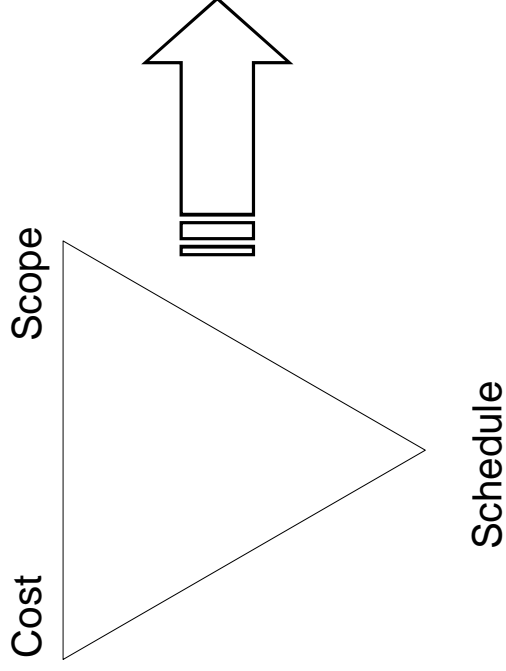
# The “Agile” Iron Triangle



Traditional Iron Triangle



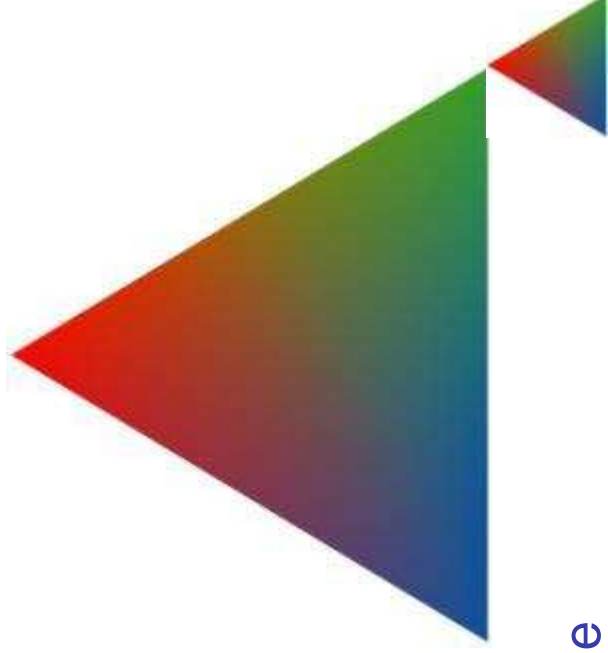
Agile Iron Triangle



# The Agile Triangle

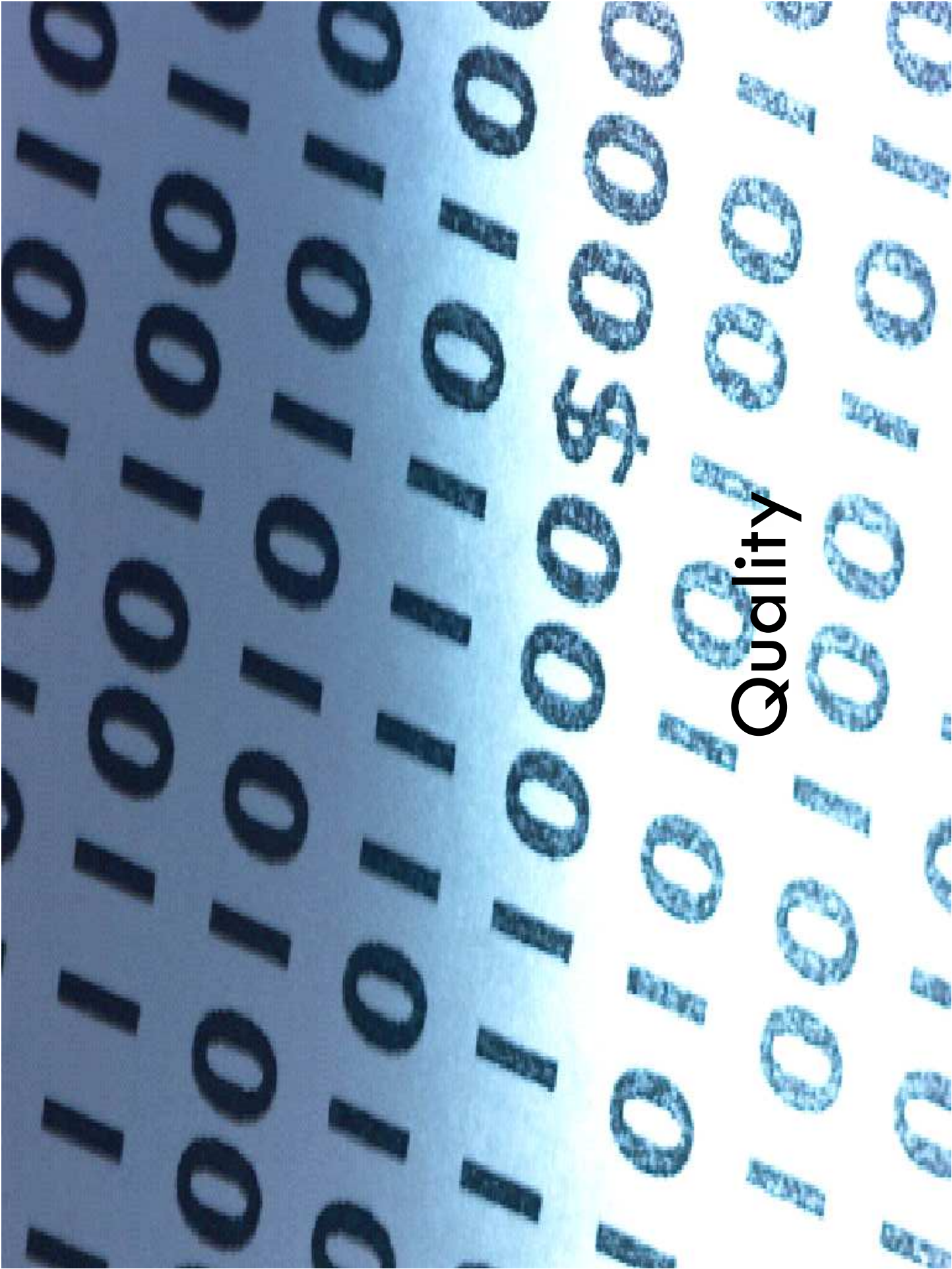


Value  
(Releasable Product)



Quality  
(Reliable, Adaptable  
Product)

Constraints  
(cost, schedule, scope)



Quality

# Is Quality Important?

**“A cumulative defect removal rate of 95% on a project appears to be a nodal point where ... benefits accrue.”**

**—Capers Jones, Applied Software Measurement (2007)**



# Scientific Instruments Co.

- Overhaul the entire product development process
- Results from 6 before- and 6 after-Agile projects

	Previous Performance	Current Performance	Percent Improvement
Project Cost	\$2.8 Million	\$1.1 Million	-\$1.7M (-61%)
Project Schedule	18 months	13.5 months	-4.5 mo (-24%)
Cumulative Defects	2,270	381	-1889 (-83%)
Staffing	18	11	-7 (-39%)

Source: Michael Mah, QSM Associates

# Why is Technical Quality Important?



- The Impact of code quality on testing
- Error Location Dynamics
- Error Feedback Ratio
- Technical Debt

# The Impact of Code Quality on Testing

Development: 10 days,  
4 people, 4 KLOC,  
1 d/KLOC



Development: 10 days,  
4 people, 4 KLOC,  
15 d/KLOC

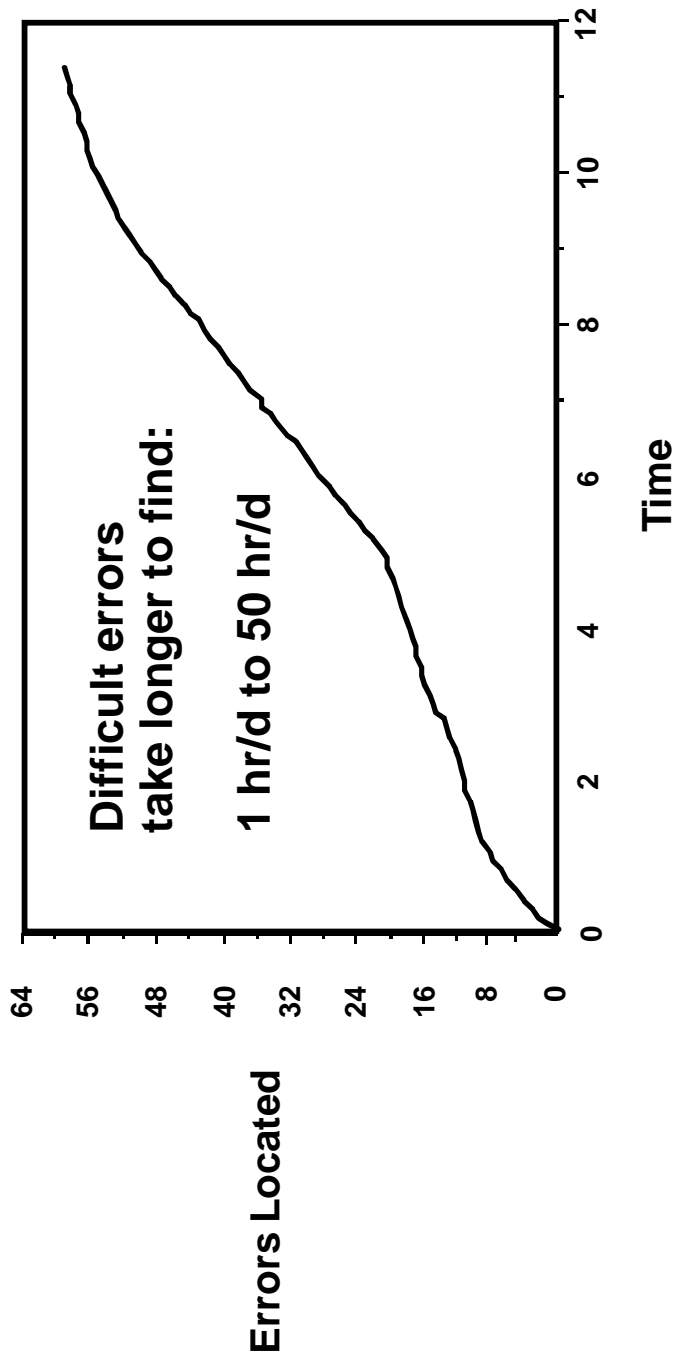
How long to test? Assume  
 $\frac{1}{2}$  day to find & fix per defect.

Test time=  
2 days

Test time=  
30 days

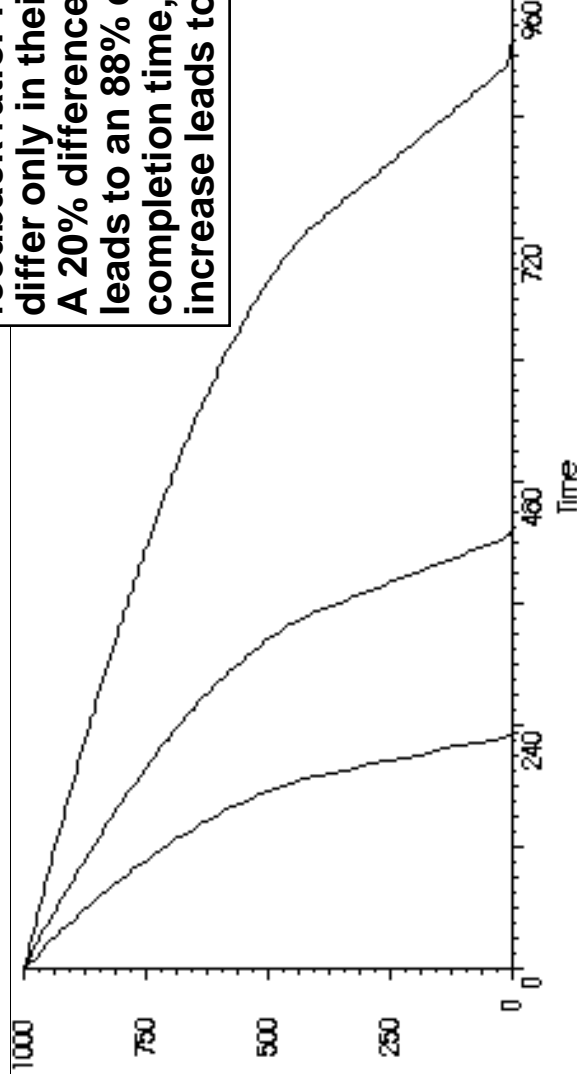
Outcome: no time to finish testing,  
technical debt increases!

# Error Location Dynamics



# Error Feedback Ratio

The time to finish removing errors is critically dependent on the error feedback ratio. The three simulations differ only in their feedback ratios. A 20% difference in feedback ratio leads to an 88% difference in completion time, but the next 10% increase leads to a 112% increase.

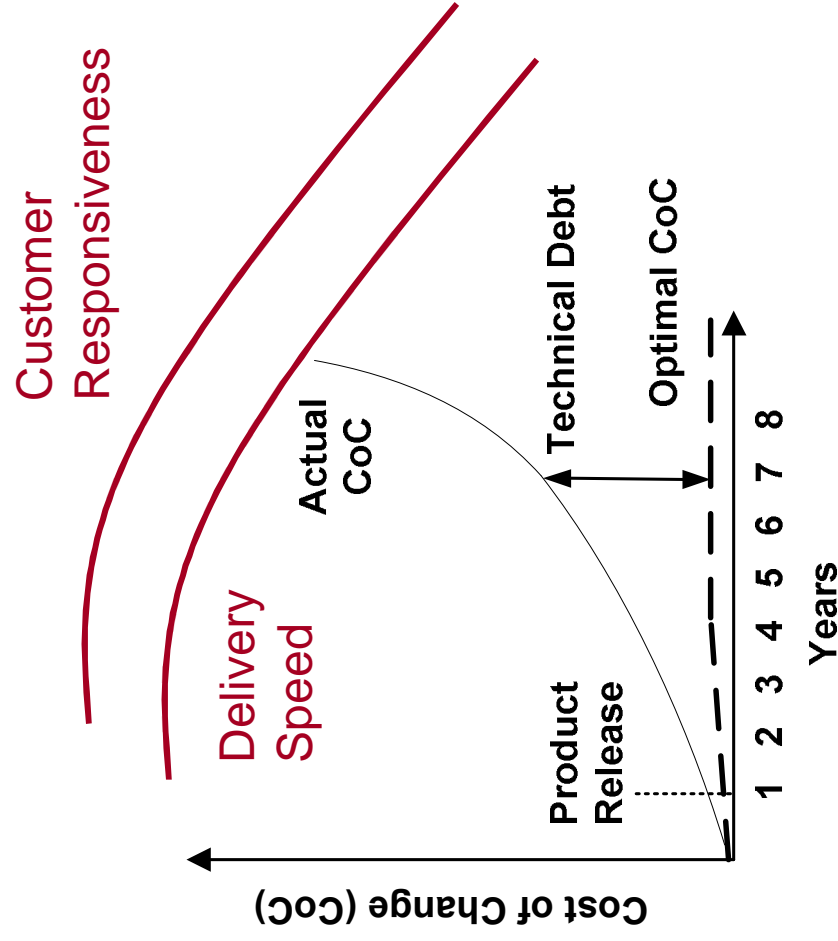


**ERROR FEEDBACK:** Errors put into a system when attempting to correct other faults.

**ERROR FEEDBACK RATIO:** The number of problems created per fix.

**EFR = ERRORS CREATED / ERRORS RESOLVED**

# Too Much Debt (Technical Debt)



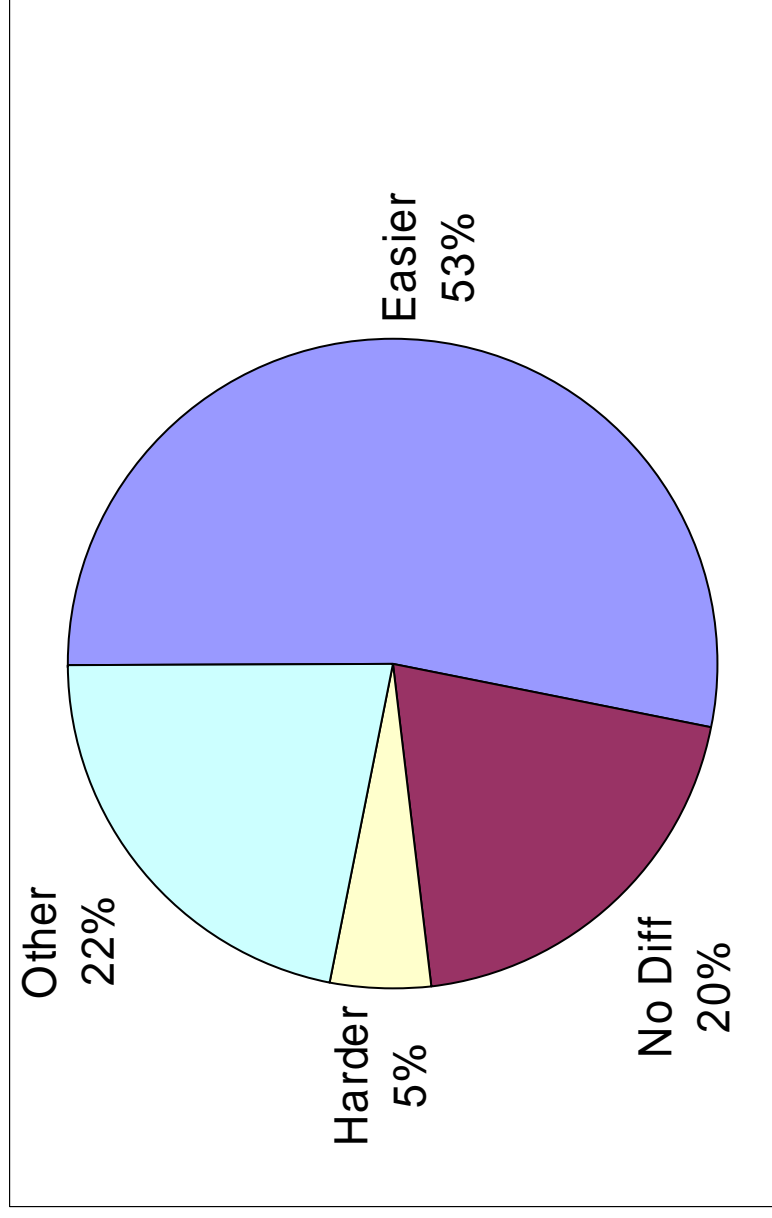
□ Once on far right of curve, all choices are bad ones

1. Do nothing, gets worse
  2. Replace, high cost/risk
  3. Incremental investment
- Estimating nearly impossible

# Agile Virtuous Cycle

- Technical agility creates a virtuous cycle of ever higher quality code and tests.
- Improves schedules.
- Reduces costs.

**Cutter Study: Agile-developed products are easier to support (maintain)**

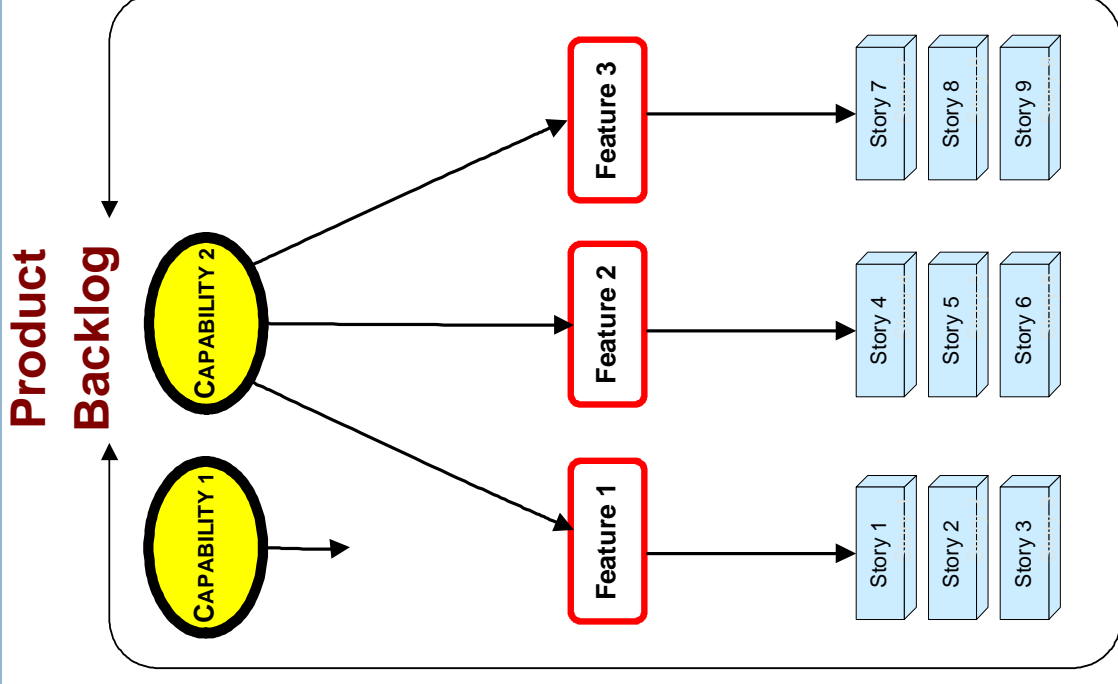


Value



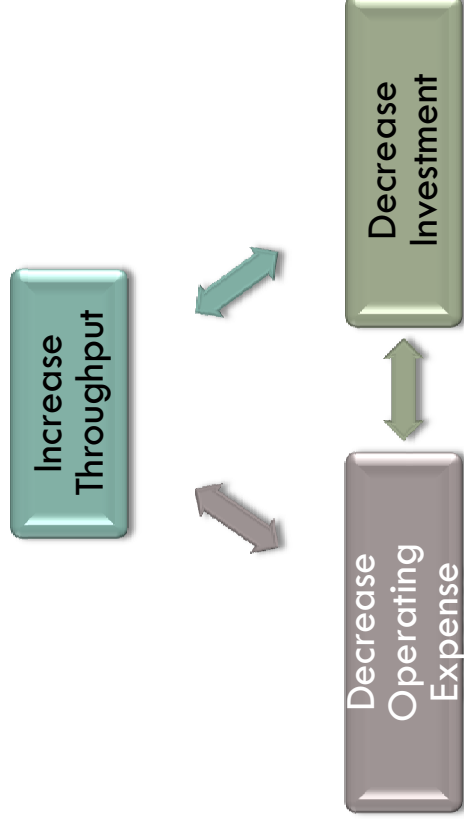
# Reducing Cost and Scope

Paul Young, VP Business  
Capabilities &  
Integration,  
MDS Sciex.



# Value Creating Opportunities

Increase: Productivity, Throughput, Flow, Highest Value Work, Profit, Growth, Share, Retention, Loyalty, Satisfaction, ROI, Efficiency, Cash Flow, Quality, Future Value, Visibility...

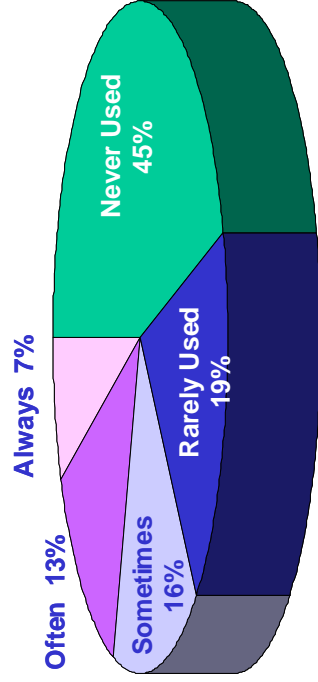


Decrease: Marginal Value Work, Cost, Time/Effort, Risk, Complaints, Turnover, Conflict, Waste (excess WIP/Inventory, waiting, rework, defects, technical debt, defects)

Improve: Engagement, Morale, Processes, Services, Collaboration, Information Flow, Quality, Loyalty, Talent (Skills), Image, Reputation, Value

Source: Pat Reed

# Scope is a Poor Progress Measure



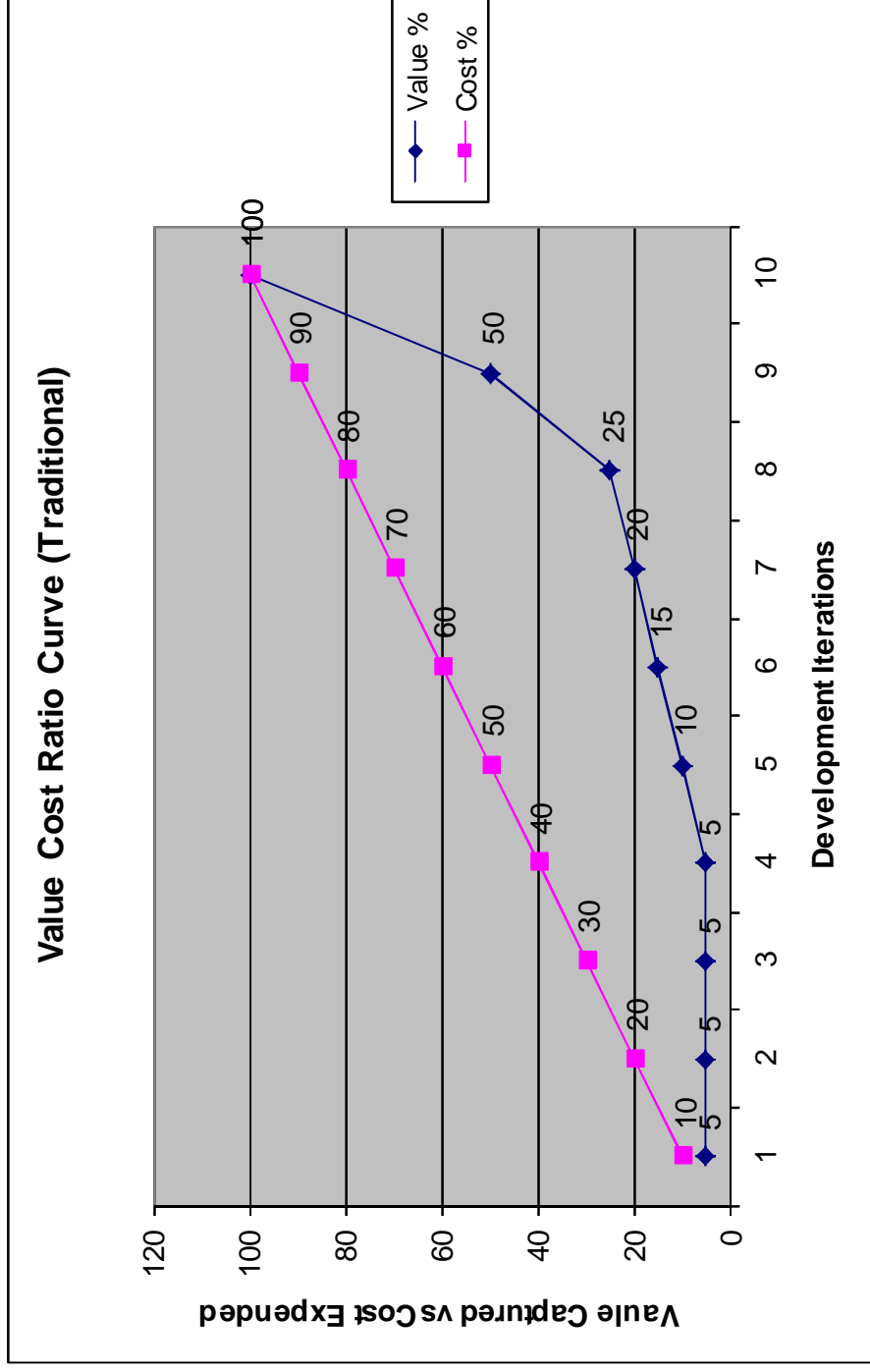
**Never or Rarely  
Used: 64%**

*Standish Group Study,  
reported by CEO Jim Johnson,  
XP2002*

2% of code used as written  
\$35 billion DOD Software  
*Crosstalk Journal 2002*

< 5% of code used  
Commercial Software,  
400 projects over 15 years  
IEEE conference 2001

# Traditional Value Curve

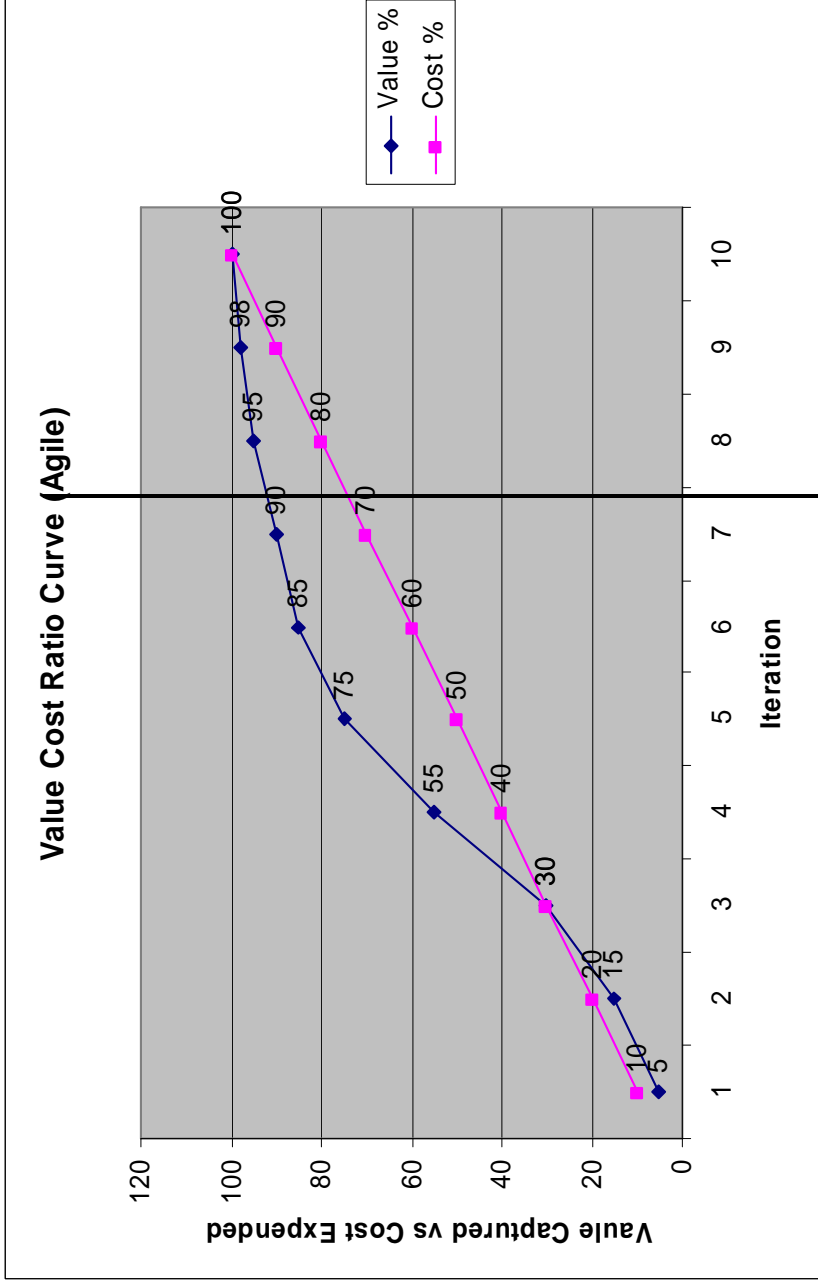


# Agile Value Curve



## Strategies

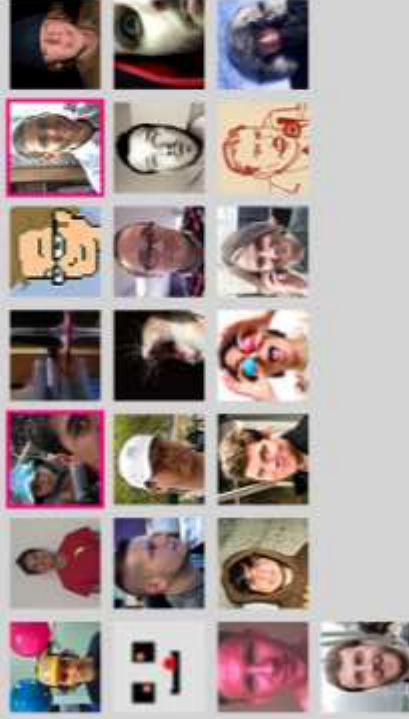
- Most valuable first
- Evolve features
- Determine right cut-off



Where is the right cut-off point?

# Continuous Delivery

## FEATURING



*Flickr was last deployed 26 minutes ago, including 8 changes by 3 people.*

*In the last week there were 47 deploys of 364 changes by 19 people.*

# Reducing Marginal Functionality

Capability					
Feature 1		Feature 2		Feature 3	
Story 1	Story 2	Story 3	Story 4	Story 5	Story 6

## Marginal Value

Simplest

Agent processes a special retail sale.

2

Basic

Agent processes a special retail sale.

4

Expansive

Agent processes a special retail sale.

8

# Value Engineering

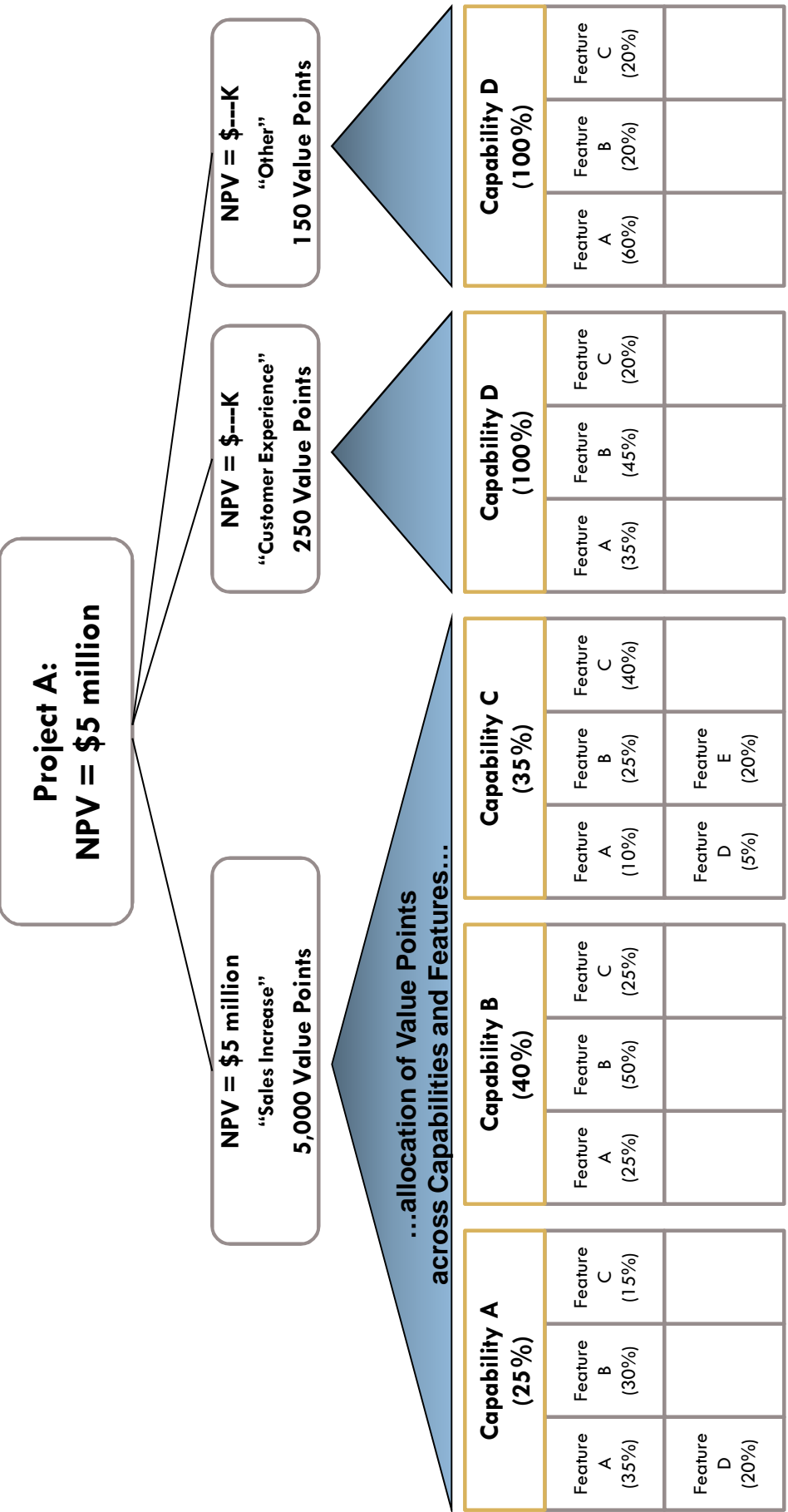
We need to understand both Value and Cost at the Capability/Feature level.

	Value	Cost
<b>Portfolio</b>	Financial Business Case (NPV/IRR)	Portfolio T-Shirt Sizing
<b>Project</b>	Same as above	Inception - Revised Cost Estimate Iterative Development - Monthly Forecast
<b>Capability</b>	<p><b>Top Down -- Allocation of Value</b></p> <p><b>Decision Making Sweet Spot</b> Where we want to start/continue to make better informed Value Engineering Decisions</p> <p><b>ROI = Value/Cost</b></p>	<p><b>Bottoms Up -- Calculation of Cost</b></p>
<b>Feature</b>		
<b>Story</b>	MoSCoW or other prioritization method	Story Points (3,5,8)

Source: Pat Reed

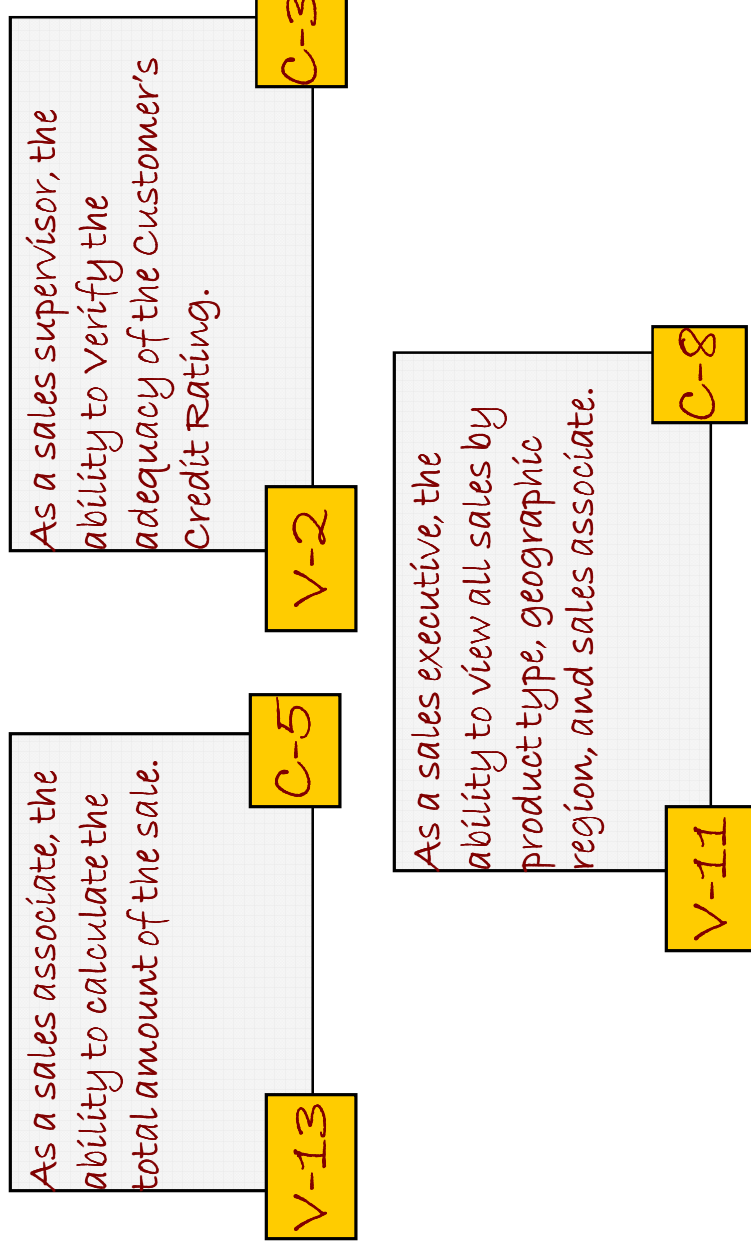
# Value Point Assignment and Allocation

Value Points get allocated across all Capabilities and Features based upon their relationship to individual Business Value Drivers:



Source: Pat Reed

# Stories with Value Points

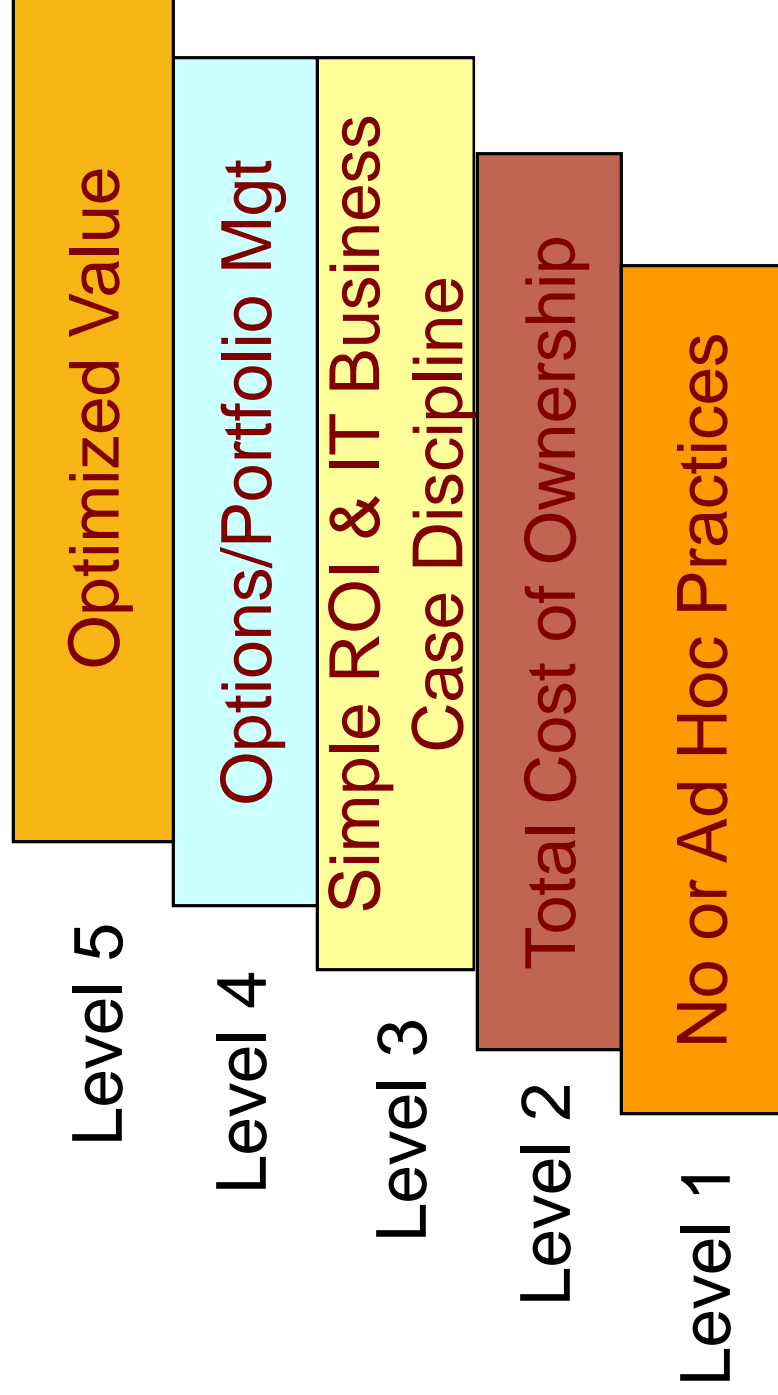


**Story Points are a calculation of cost.**  
**Value Points are an allocation of revenue.**

***“If you don’t have time to estimate value, we don’t have time to estimate cost!”***



# Business Value Maturity Model (Intel)



# The GAP: Business Value Dials

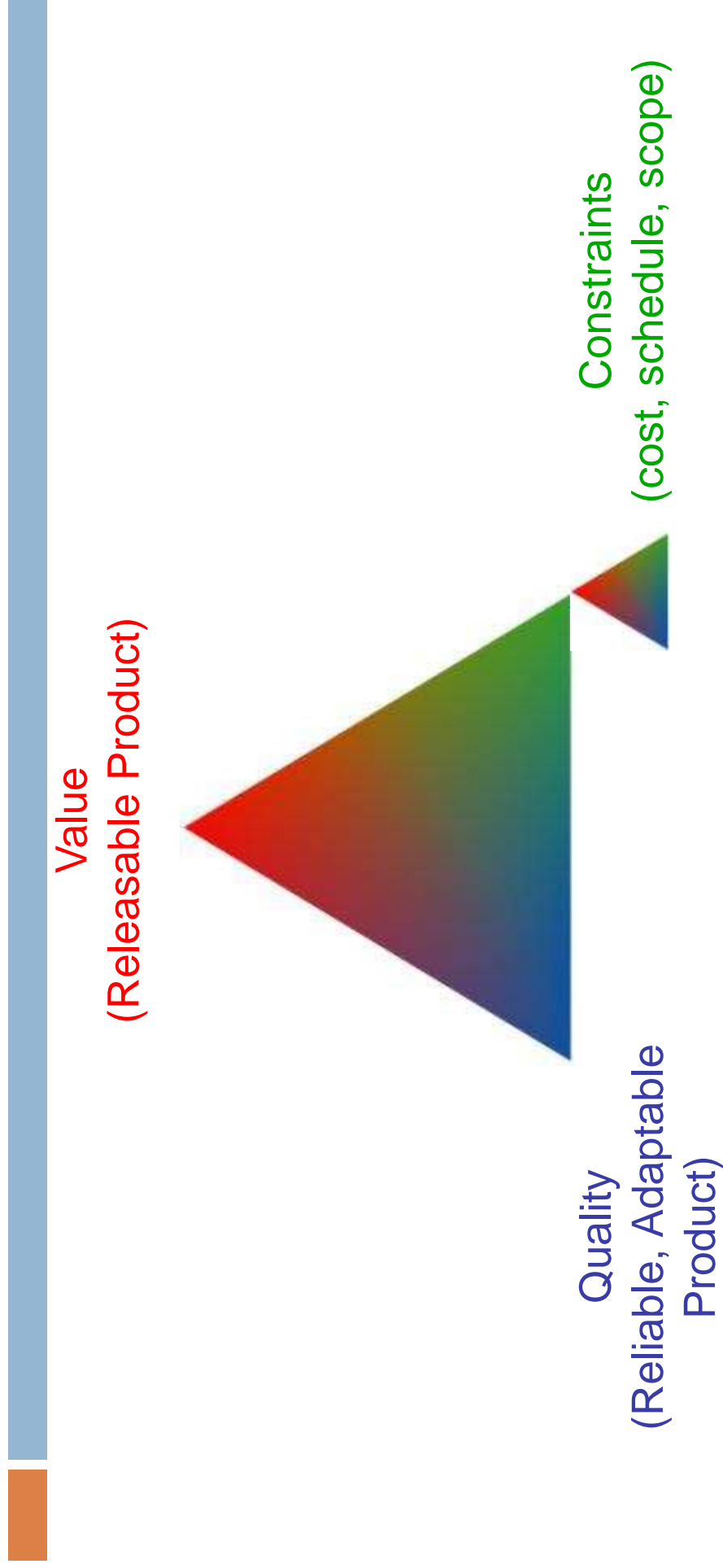
- Headcount Reduction (# of H/C reduced or avoided) x (Average burden rate for region & job type)
- Headcount Productivity (Number of employees affected) x (Time) x (Avg. burden rate) x (50%)
- Headcount Turnover (33% of annual burden rate/region/job type) x (# H/C turnover avoided)
- System End-of-Life (EOL) Cost of maintaining EOL'd system
- Hardware/Software Avoidance Total cost of the H/W or software avoided
- Unit and Other Cost Avoidance Total of actual costs avoided
- Fulfillment Center Optimization (Value of product) x (Volume increase)
- Waste Reduction Total value of waste reduced or avoided
- Risk Avoidance (Value of risk) x (Probability of occurrence)
- Time-To-Market (Value of increased market segment share) x (# weeks accelerated to market)
- Open New Markets (Increase volume) x (Average selling price)
- Optimize Existing Markets (Increase volume) x (Average selling price)
- Cross-Selling (Increase volume) x (Average selling price)
- Direct Revenue Total amount of revenue generated by projects

Adapted from Intel

# Value and Priority



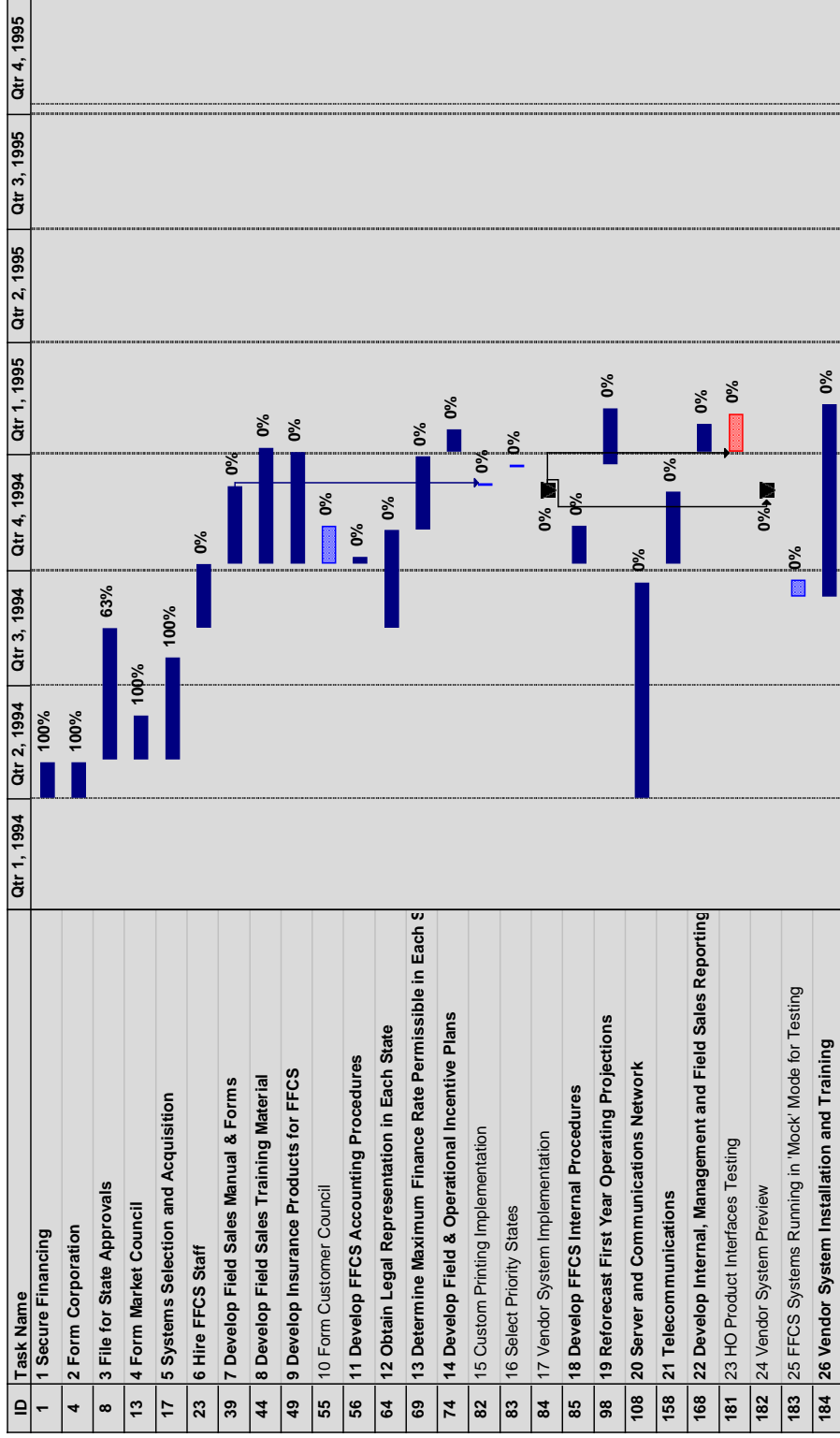
# The Agile Triangle



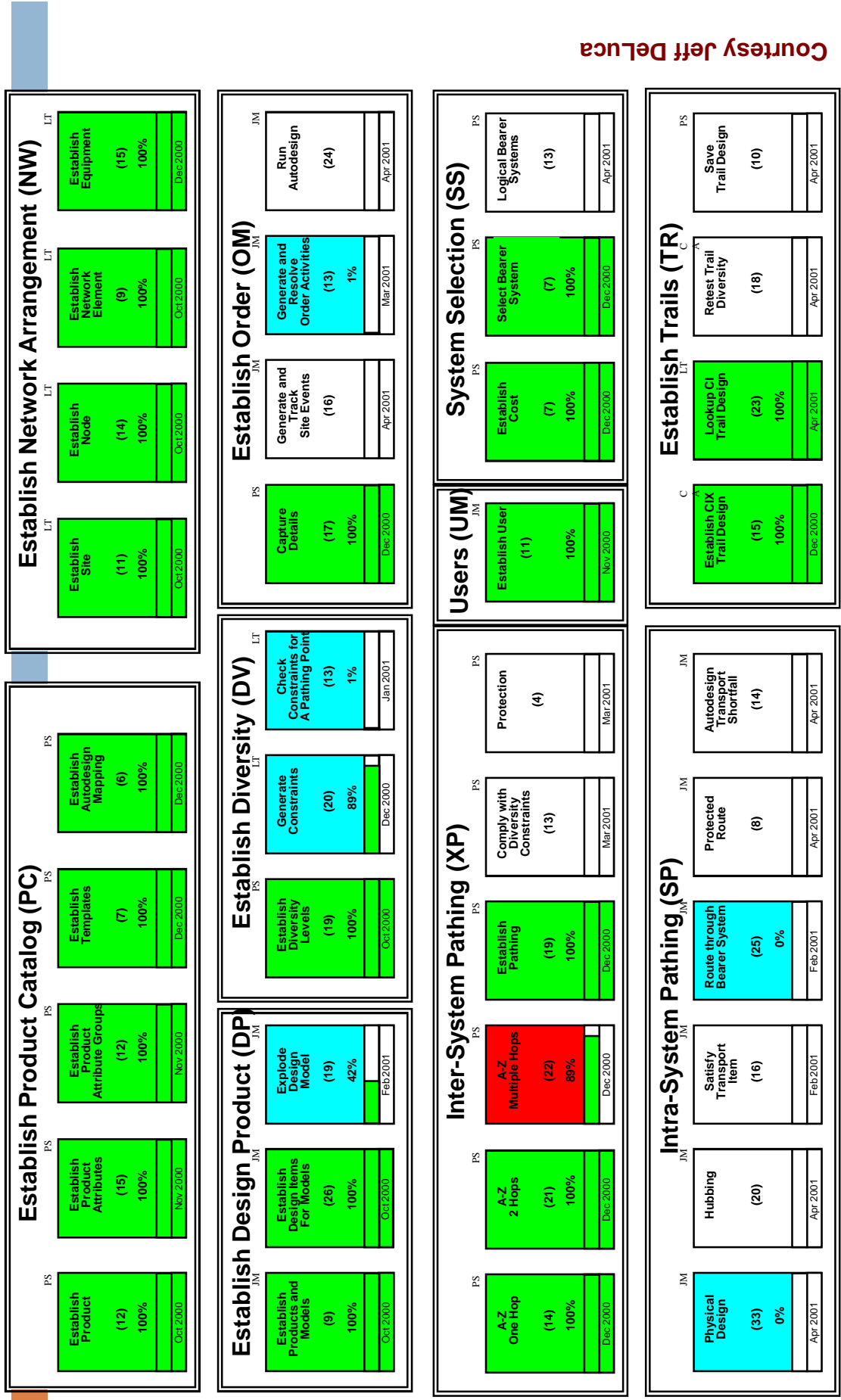
A traditional project manager focuses on following the plan with *minimal* changes, whereas an agile leader focuses on “*adapting successfully to inevitable changes*”.



# Traditional Gantt Chart — What Does It Emphasize?



# A Parking Lot Diagram — What Does It Emphasize?



**KEY:** █ Full Completion █ Attention █ Work In Progress █ Progress Bar

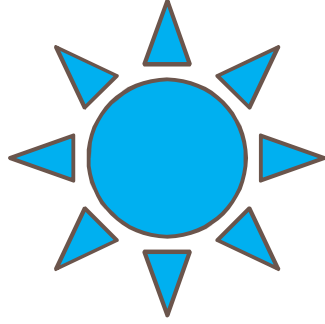
Courtesy Jeff Deluca

# Strategic Questions

- Why couldn't we release this product today?
- What is our value-cost ratio?
- What is the product quality?
- Are we within acceptable constraints?



# Albert Einstein



No problem can be solved from the same  
level of consciousness that created it.



Thank  
You!



QUESTIONS?

## Next Month, April 19th

- Seven Deadly Sins of Project Management with  
Mike Cohn

Thanks to our sponsor

