

# Using the TSP at the End of the Development Cycle

Scott Van Eps and Rick Marshall

Beckman Coulter

September 18, 2012



# Beckman Coulter: *Reducing the Cost of Care*

Global development,  
manufacturing and  
marketing of hospital  
and critical care  
**diagnostic systems**



Used to diagnose  
disease, make  
treatment decisions  
and monitor patients.



# Beckman Coulter: *Simplify . Automate . Innovate*

Chemistry



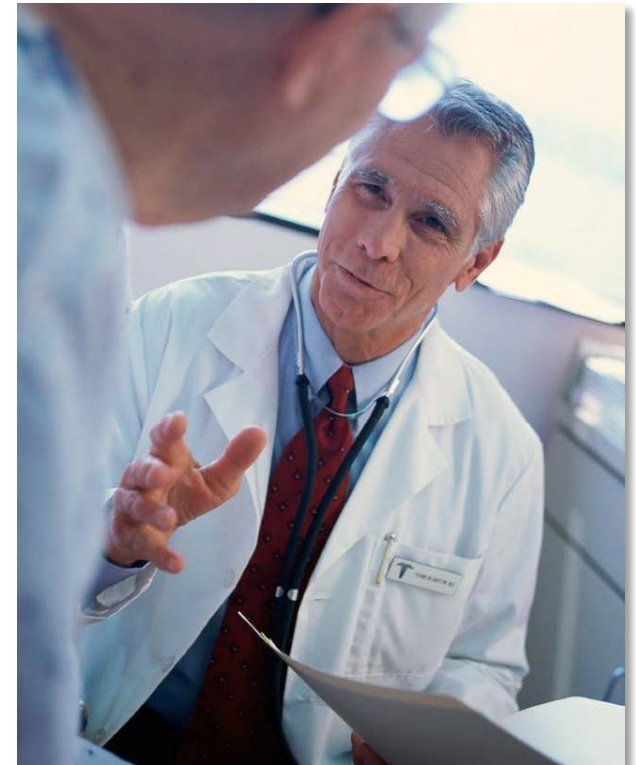
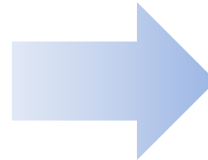
Hematology



Immunoassay



Automation



**We produce reliable test results**

For over 75 years, our products have supplied critical information to improve patient health.

# TSP at Beckman Coulter

## 2008: Search for Demonstrated Outcomes

A software process that **demonstrates:**

- Higher Quality
- Predictable schedules
- Increased productivity
- Lower Maintenance
- Good for the Engineer



# TSP at Beckman Coulter

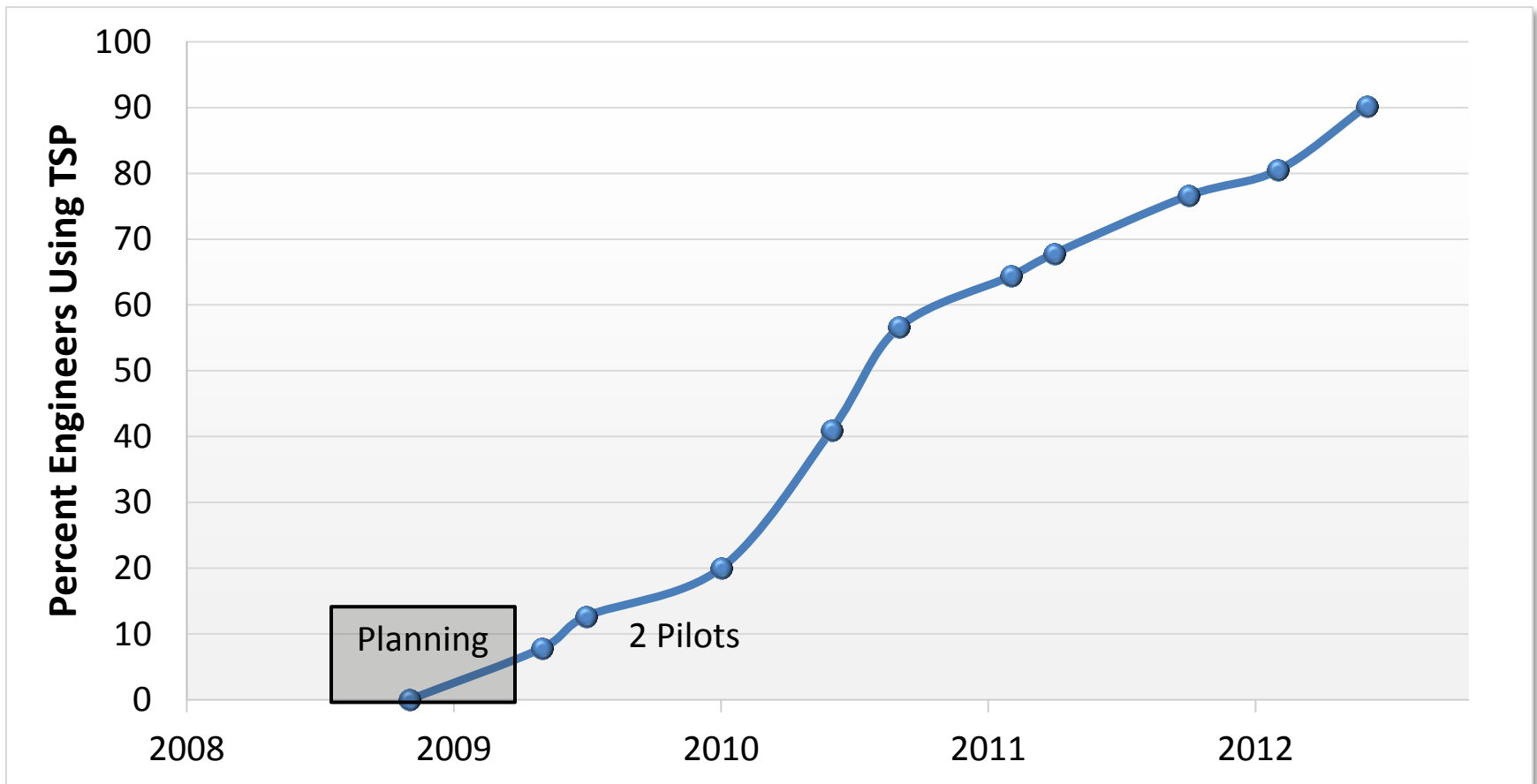
## 2008: Search for Demonstrated Outcomes

- Surveyed Engineers
- Contacted Universities
- Researched LEAN, Six Sigma, and Agile
- Met with Jim Over to understand TSP
- Discussed TSP with industry adopters
- Selected TSP
- Presented overview to engineers, project leads, functional managers
- Selected 2 pilot projects



# TSP at Beckman Coulter

In three years, achieved 90% adoption across Diagnostics R&D





# Hematology Project Status



## DxH 800

- Next-Gen Hematology
- Introduced in 2008
- *Customers waiting for next release*



## Slide-Maker-Stainer (SMS)

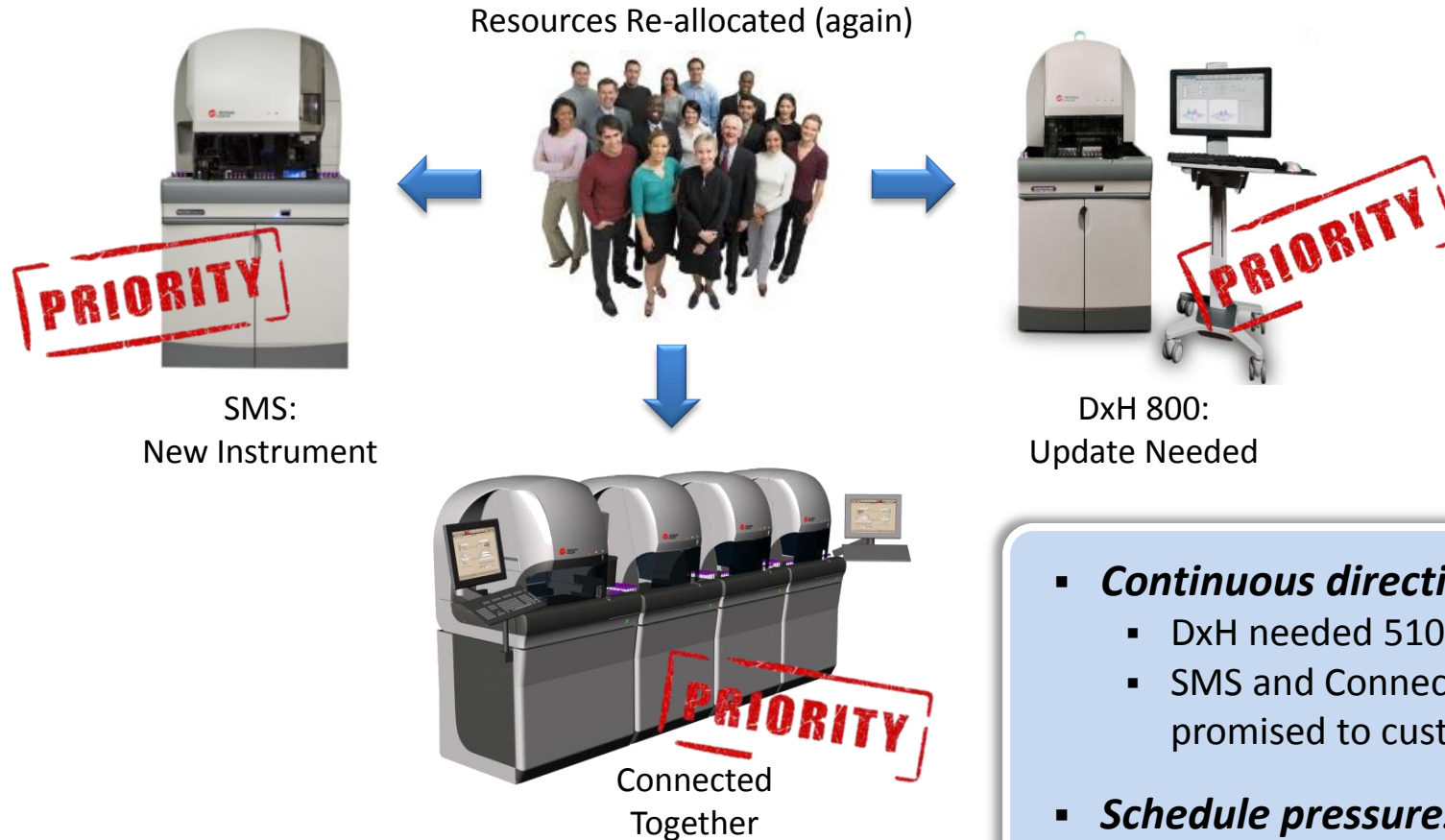
- Next-Gen Slide Maker in development
- *Customer waiting for initial release*



## DxH 2401 Connected System

- *Market Need*

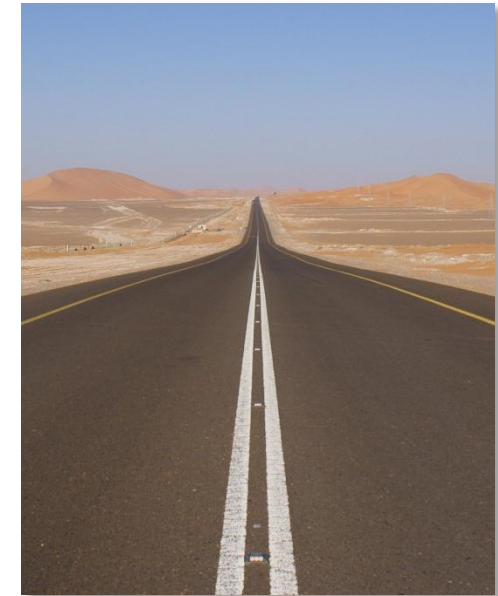
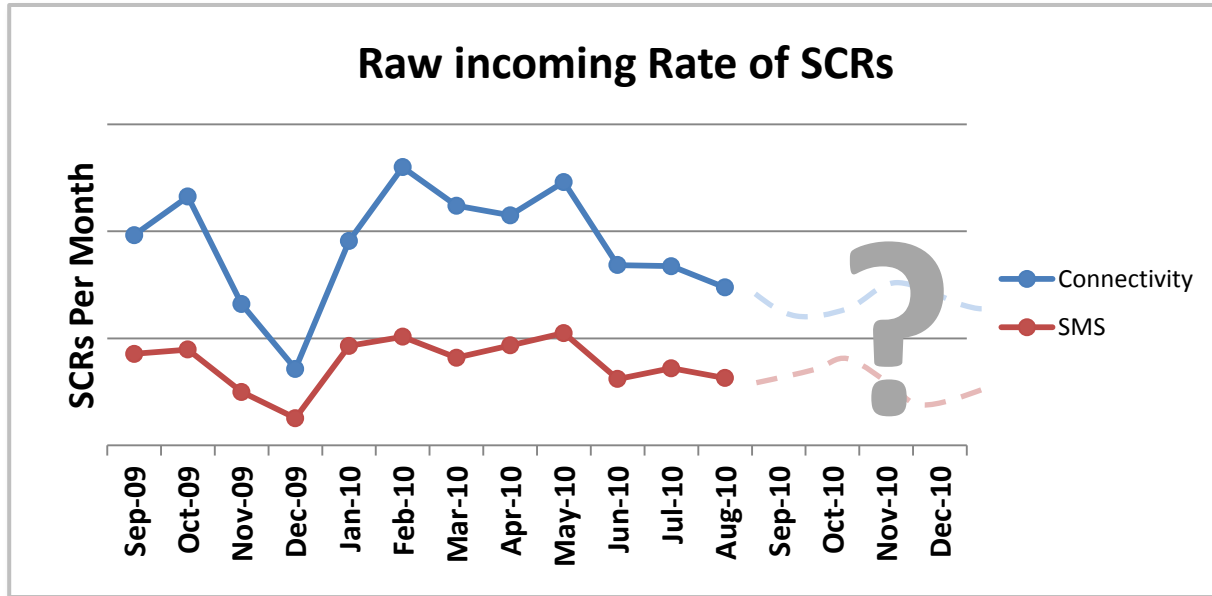
# The Situation



- **Continuous direction changes**
  - DxH needed 510(k) for FDA
  - SMS and Connectivity was promised to customers
- **Schedule pressures**
  - No time for inspections
  - Low staff moral (potential death march)



# The Situation



- Large Code Base:
  - DxH and SMS code base were each over 3,000 KLOC
  - Each had greater than 1,100 KLOC Added/Mod./Deleted
- SMS 3-4 years in development
- DxH updates over 2 years in development
- In System Test for 8 months

Incoming Software Change Requests (SCRs) not decreasing

# The Approach: *Estimate Effort & Set Priority*

## ***When will the projects get done?***

How many defects are left?



Estimate the effort left on each project



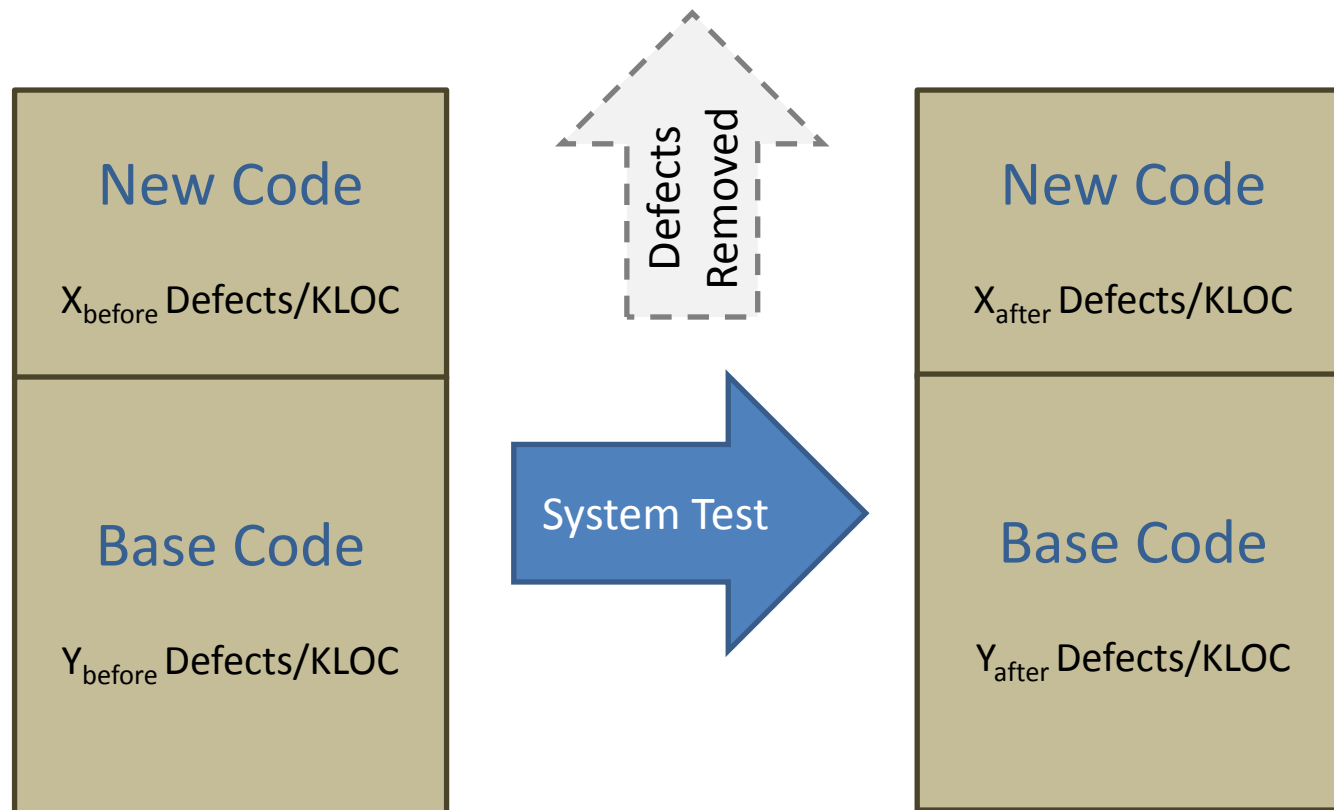
Based on resourcing options,  
set priority and develop top level plan



# The Approach: *Estimating Defect Left*

## Technique #1

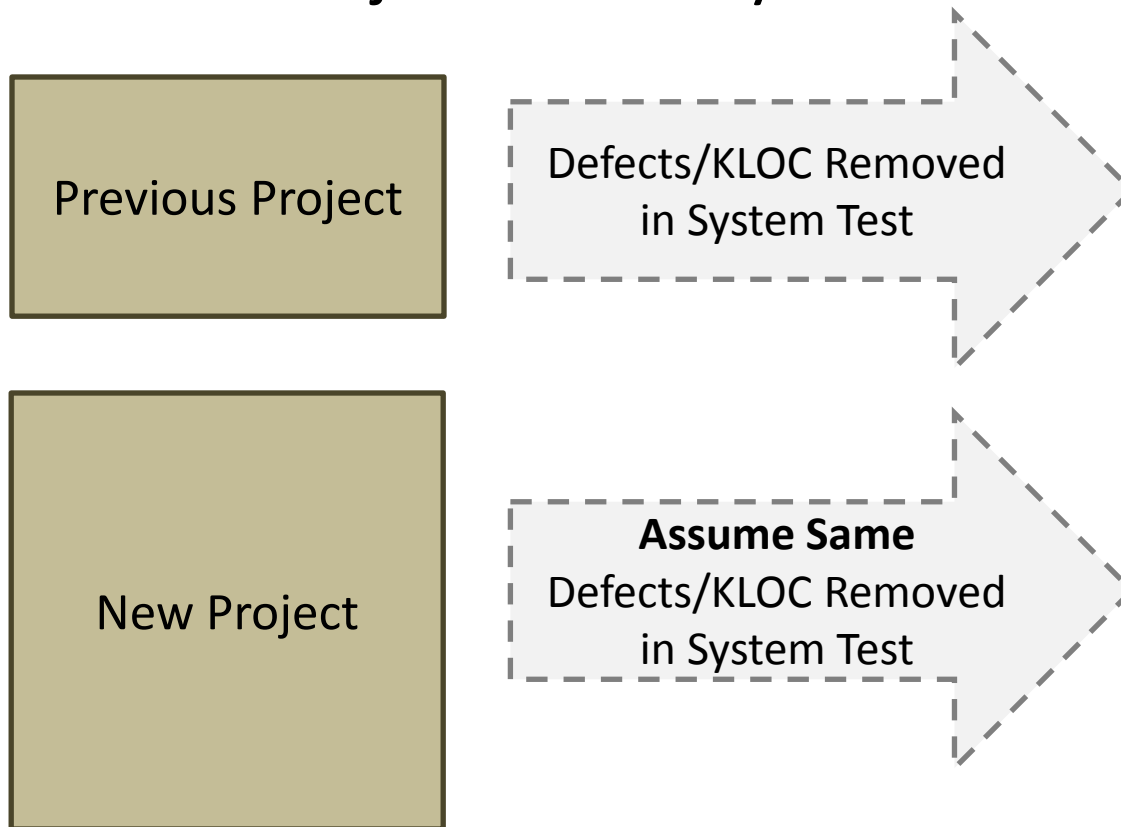
Model reduction in defects/KLOC through System Test



# The Approach: *Estimating*

## Technique #2

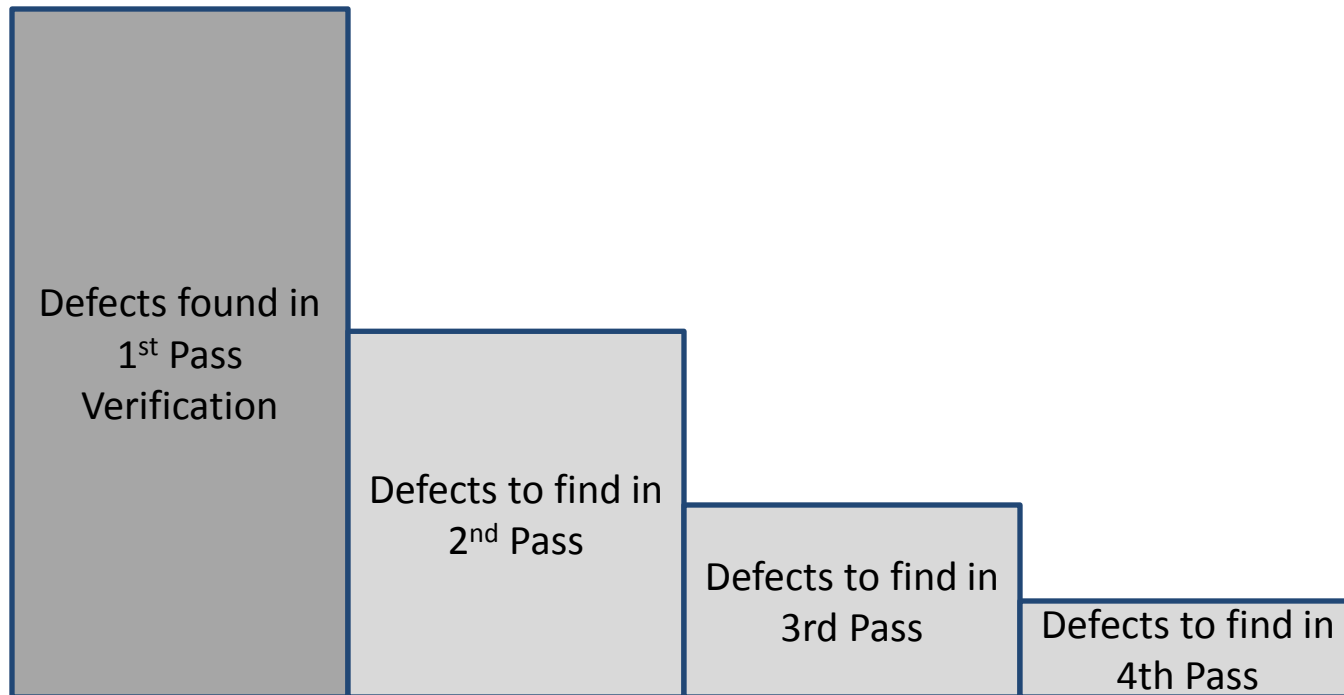
### Use Previous Projects as Proxy



# The Approach: *Estimating*

## Technique #3

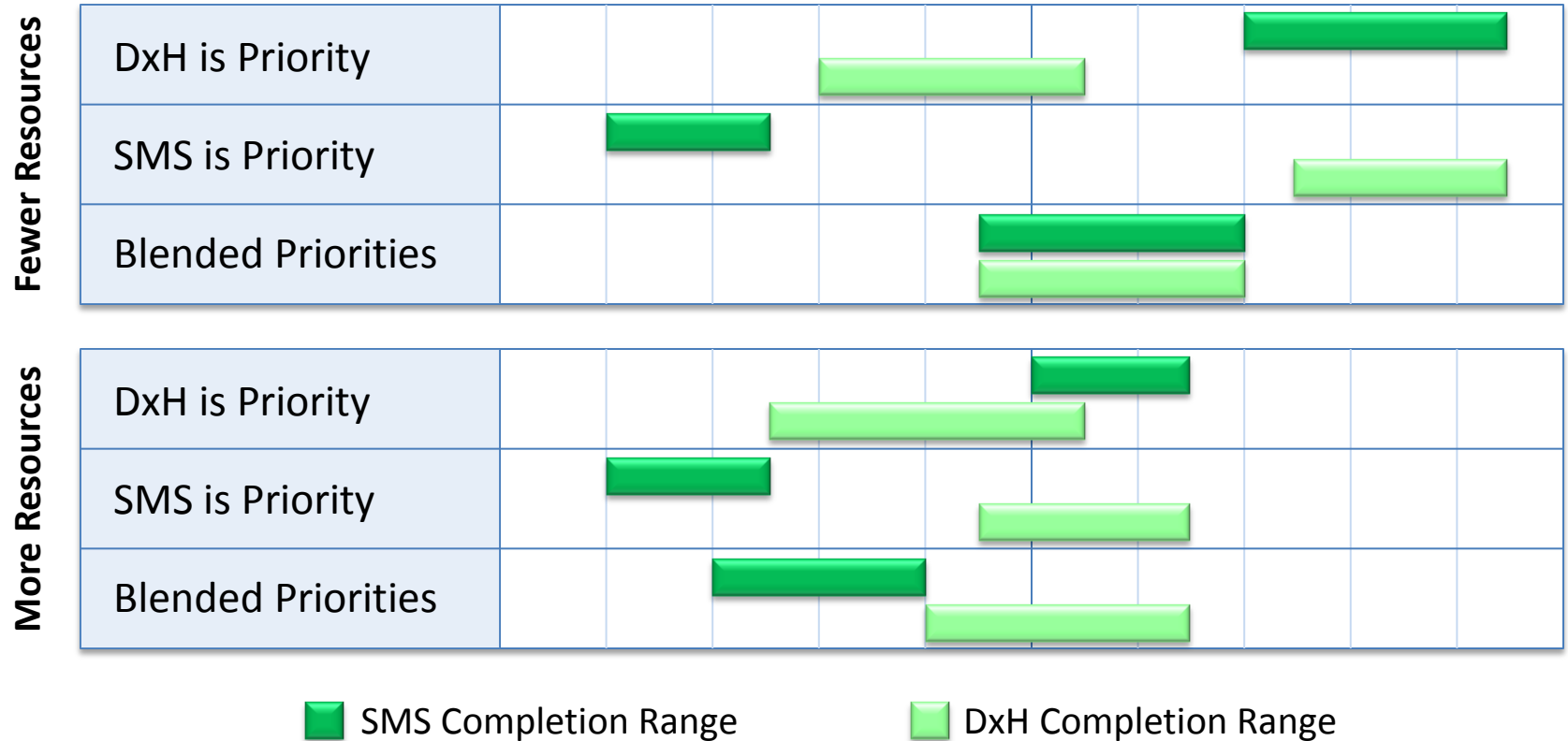
Assume 50% defects found with each pass of verification



## The Approach: *Decide on Priority and Resourcing*

## Developed models for management

- Total resources available
- Priority of each project
- Use best-worse number of defects to estimate ranges





# The Approach: *Workshop for Developers*

## Cycle 0

- Message from Management: More people, more time, a better process
- Formed Workshop Coordination Team
- Introduced PSP/TSP principals based on what was applicable to project in this phase
  - SEI-tailored for just-in-time training
  - Introduced inspections and personal checklists
  - Put time in schedule for inspections and checked rates
  - Cycle planning
  - Data capture for future estimation
  - Role Managers
  - Coaches
  - Weekly Meetings
  - Set expectations of no new defects into System Test
  - Manage Risks
- Trained Development and Verification teams



# The Approach: *Legacy Code*

## “Good Ideas List” Driven by Team

- Asked developers to **identify areas** of design, code, process that we should address
- Goal was to quickly stabilize and increase quality of the product
  - Gave them defect density, code churn, and Software Change Request (SCR) data
  - Team came up with over **100 ideas**
  - Team evaluated and ranked ideas. Identified early wins
  - Best ideas were **scheduled into each cycle**
- Introduced **Static Analysis Tools** to address Legacy Code



# Approach: *Future Cycles*

## Remaining Cycles

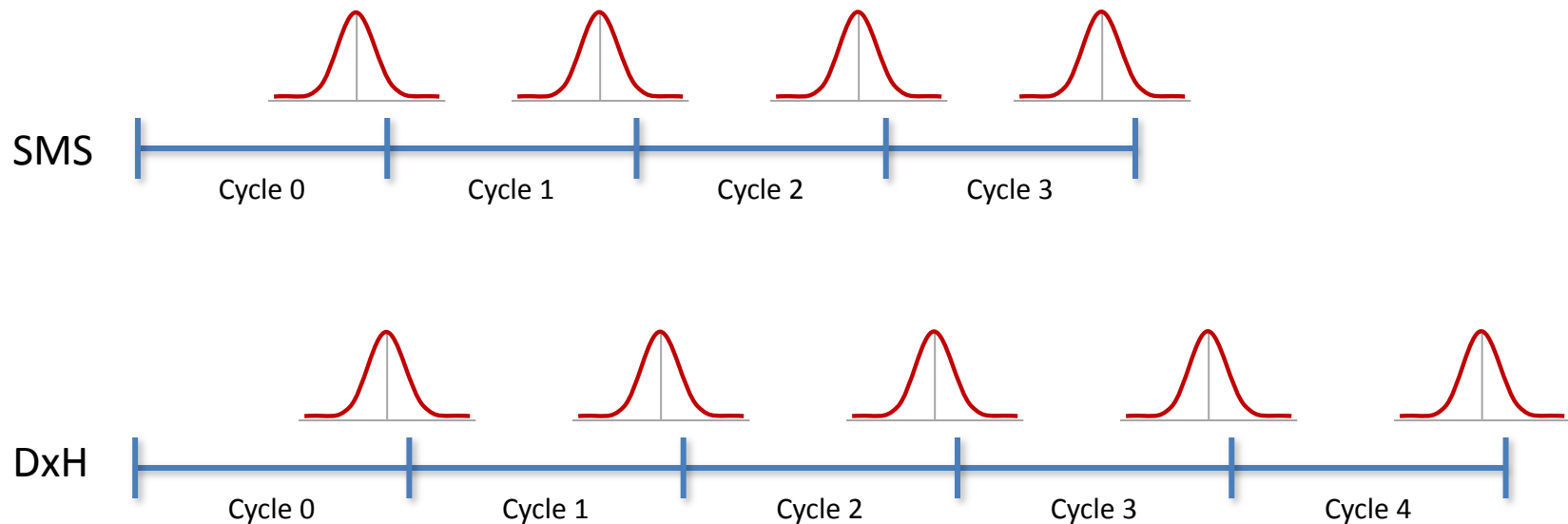


- Introduced topics as needed each cycle
  - Refined estimates with Task Hours vs. Calendar Hours
  - Used a tool to better capture WBS and estimates
  - Retrospectives after each cycle
  - Weekly Recovery Plans when needed
  - Set Goals Per Cycles based on Project needs
- Introduced Monte Carlo modeling to estimate completion date

# Implementation: *Monte Carlo Modeling*

## Monte Carlo Modeling of Triage, Development and Verification

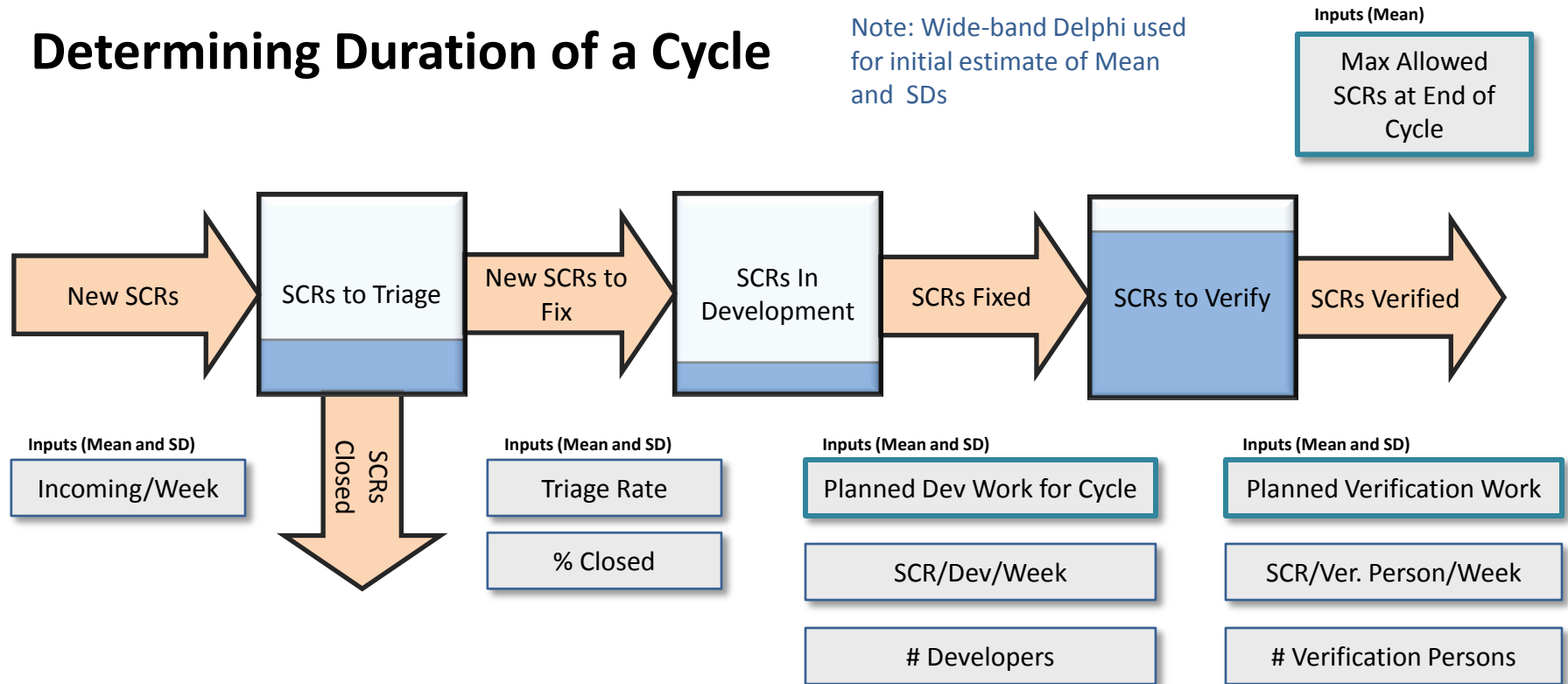
- Used to estimate cycle and project completion times throughout project
- Provided input to cycle planning to establish goals
- Balanced resources between SMS and DxH projects
- Balanced resources between Development and Verification (prevent starvation)



# Implementation: *Monte Carlo Modeling*

## Determining Duration of a Cycle

Note: Wide-band Delphi used for initial estimate of Mean and SDs



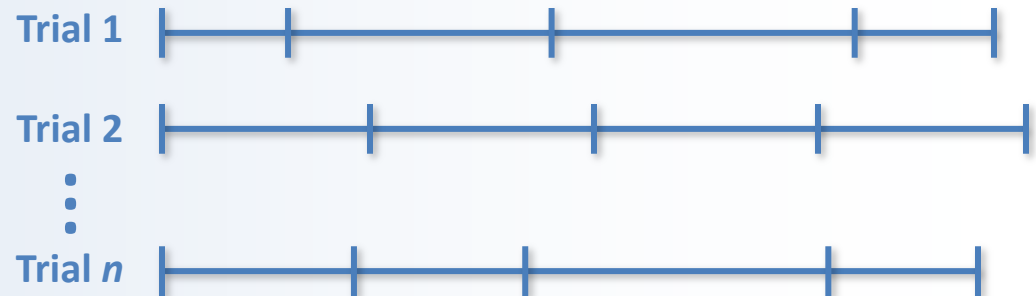
### Cycle Ended When:

- Planned Development Work for cycle completed
- Planned Verification Work for cycle completed
- SCRs to Verify < Max Allowed

# Implementation: *Monte Carlo Modeling*

Simulate  $n$  trials with random inputs based on mean and SD of variables:

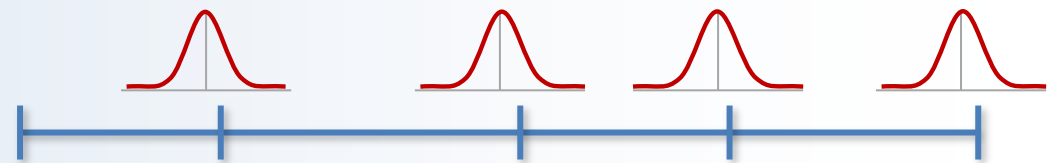
- Incoming /Week
- Triage Rate
- % Closed
- Dev Work for Cycle
- SCR/Dev/Week
- # Developers
- Protocol Work
- SCR/SVV/Week
- # Verification Persons



Combined into statistical distribution of cycles

## Model used to set Meeting 1 goals

- # Developers
- # Verification Persons
- Fixed Dev. work to do
- Protocol work to do
- # SCRs to Triage
- # SCRs to Correct
- # SCRs to Verify



Used 84 percentile of a cycle's distribution for commit dates

Replaced with Actuals after each cycle



# Results: *SMS Predictability*

Cycle 0 Launch 9/13/2010

**Cycle 0**  
Stabilization  
13 weeks

Planned: 12/13/2010  
Actual: 12/13/2010

**Cycle 1**  
Enter Design  
Validation  
14 weeks

Planned: 4/1/2011  
Actual: 3/31/2011

**Cycle 2**  
Complete all  
Protocols  
12 weeks

Planned: 6/22/2011  
Actual: 6/30/2011

**Cycle 3**  
Release Candidate  
12 weeks

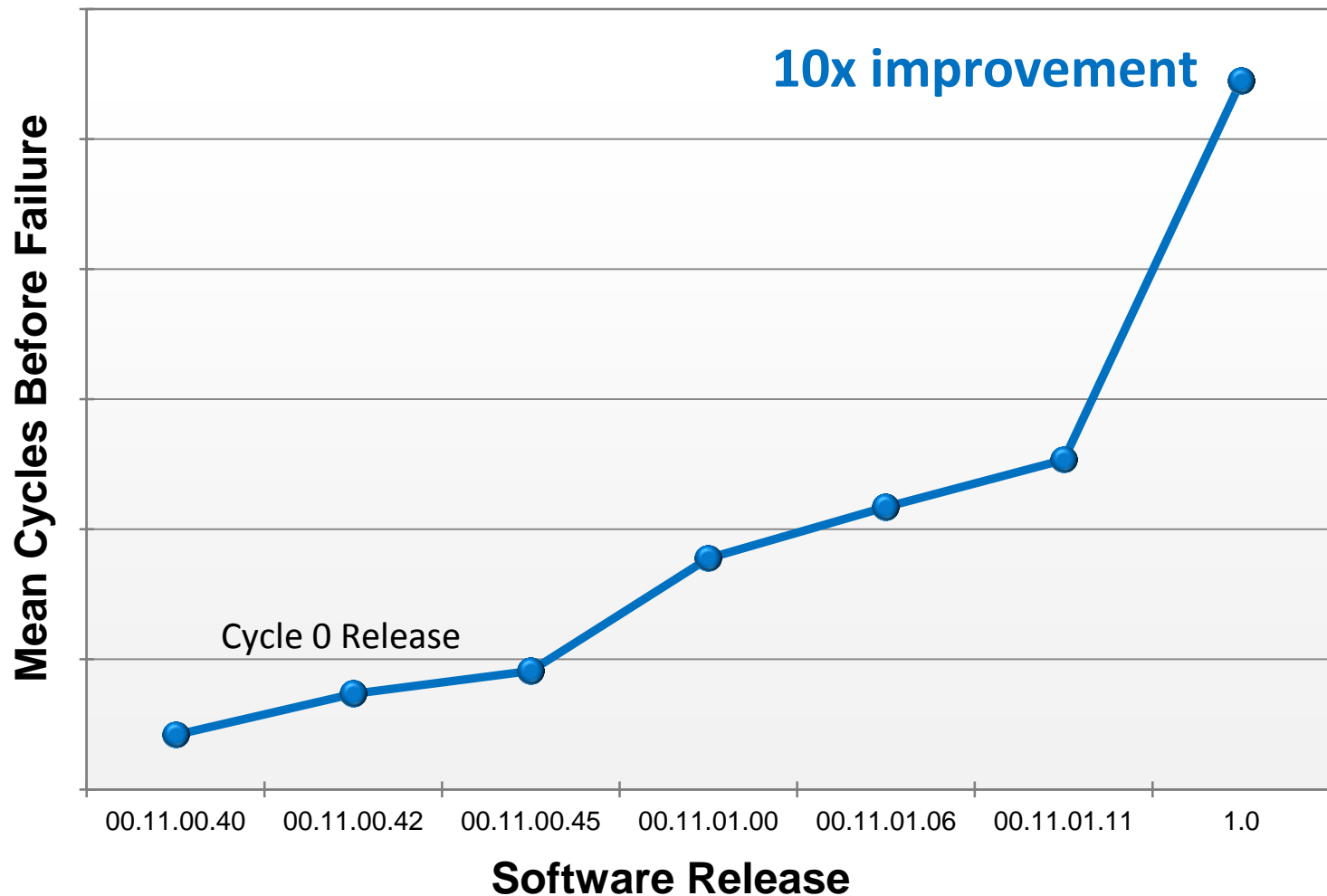
Planned: 9/14/2011  
Actual: 10/17/2011

**Schedule Predictability: 3.4%**

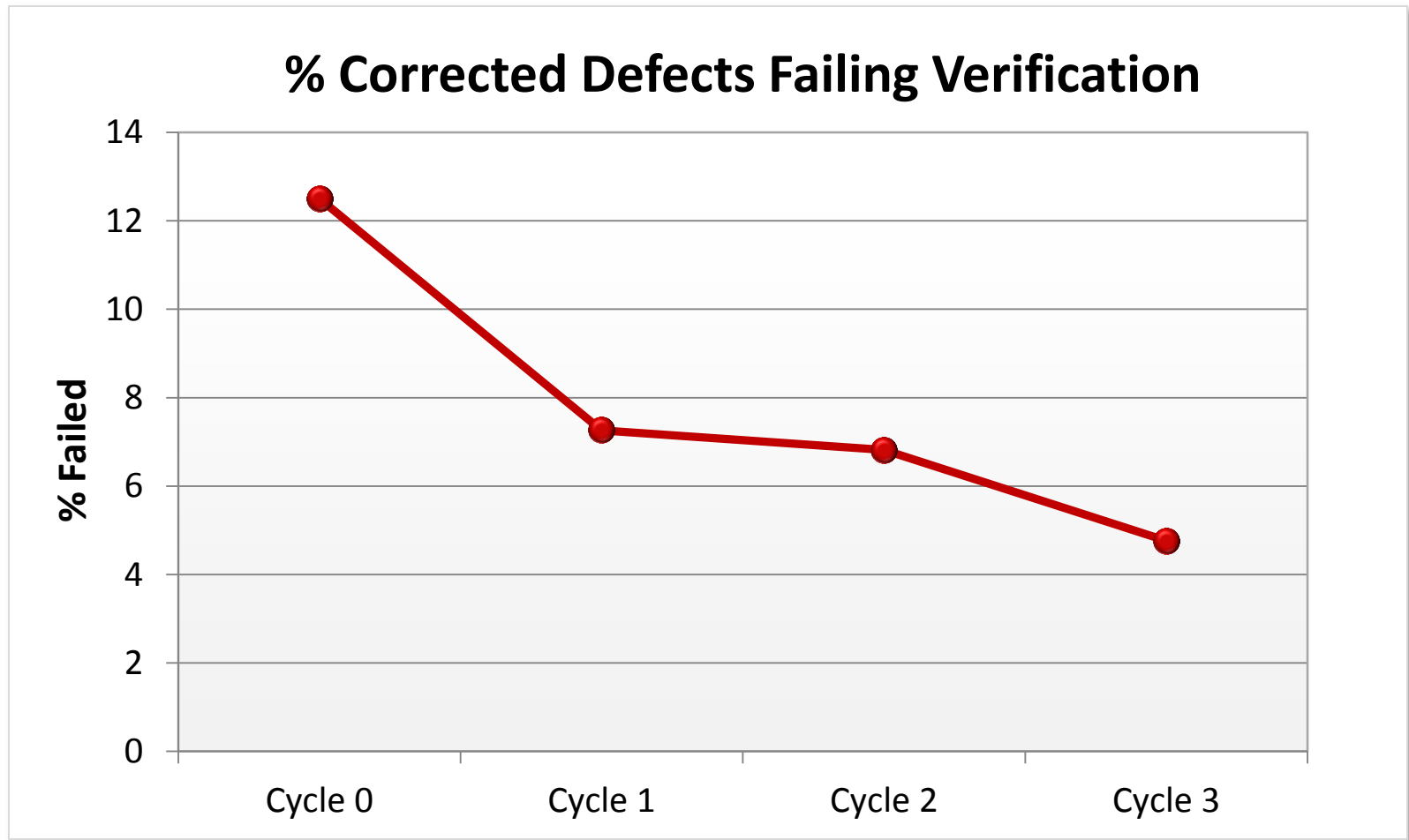


Initiate Ship  
Planned: 10/31/2011  
Actual: 11/14/2011

# Results: *SMS Reliability*



# Results: *SMS Cleaner Code*



# Results: *SMS in the Field*



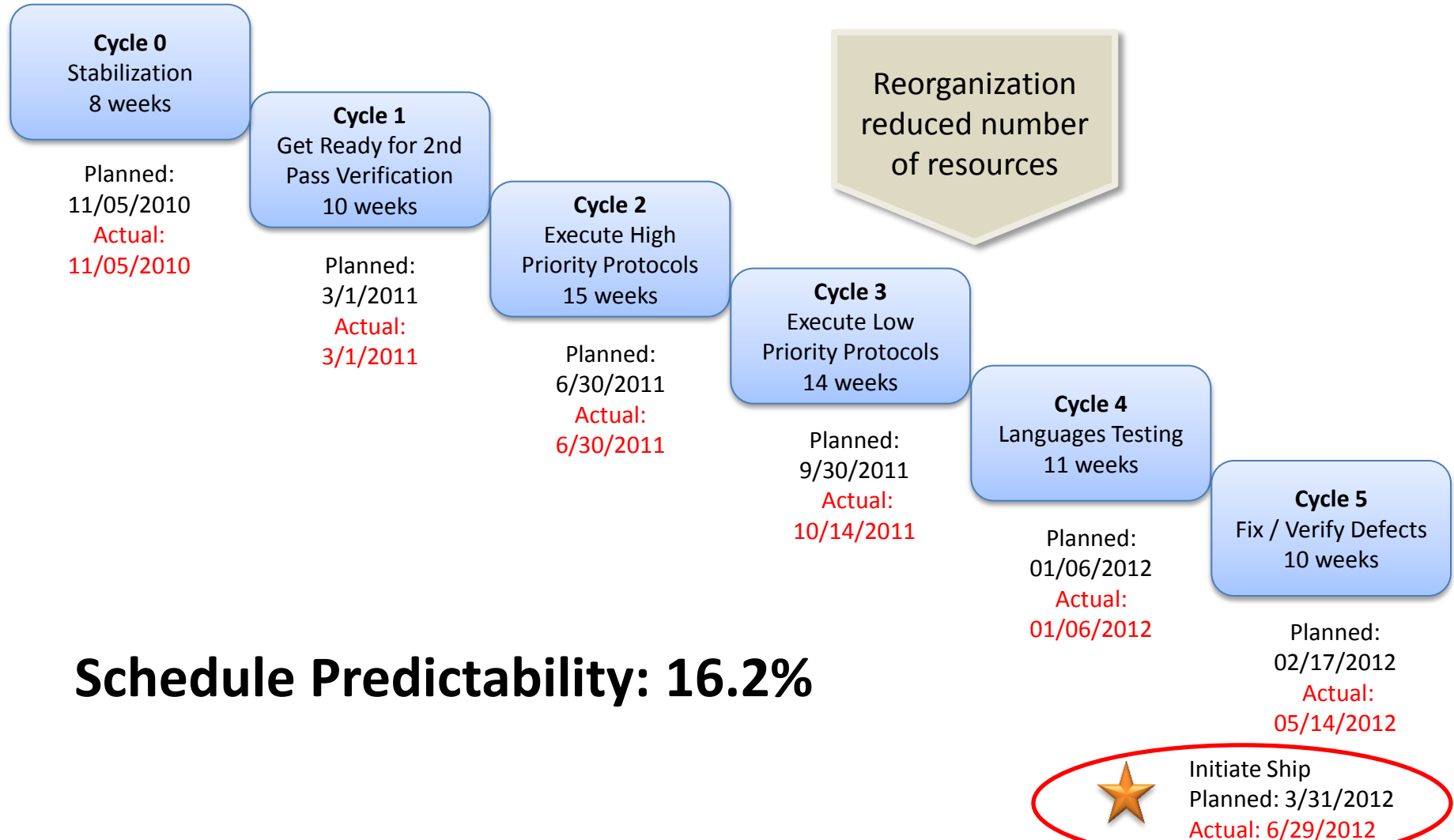
## SMS

Released: 11-14-2011  
Installs: Over 175 installed  
Results to Date: No issues causing a new release  
Very good customer acceptance!

*"The SMS is a fantastic addition to our lab workflow"*  
*"We have had very little cause for service calls"*  
*- Our Customers*

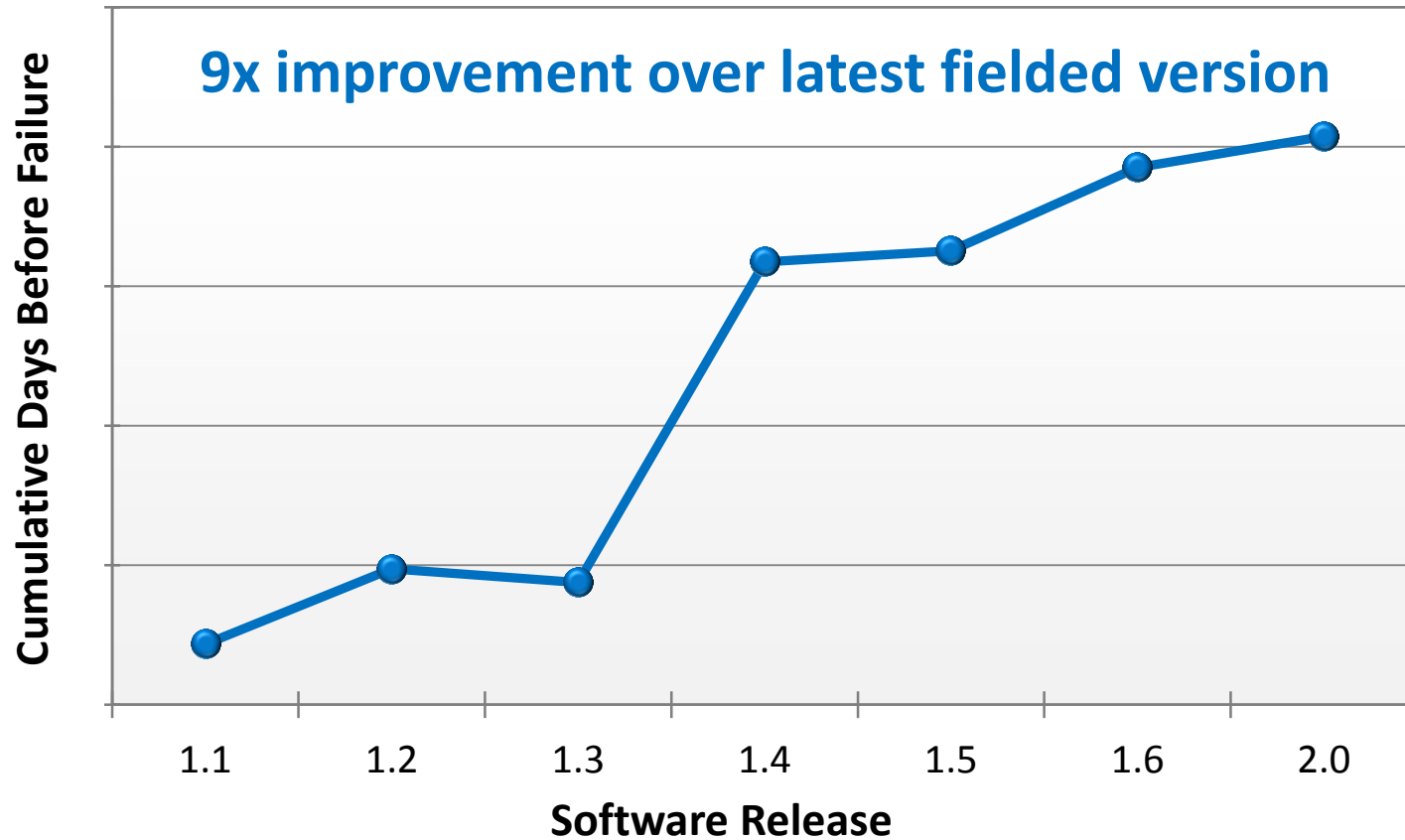
# Results: DxH 800 Predictability

Cycle 0 Launch 9/13/2010



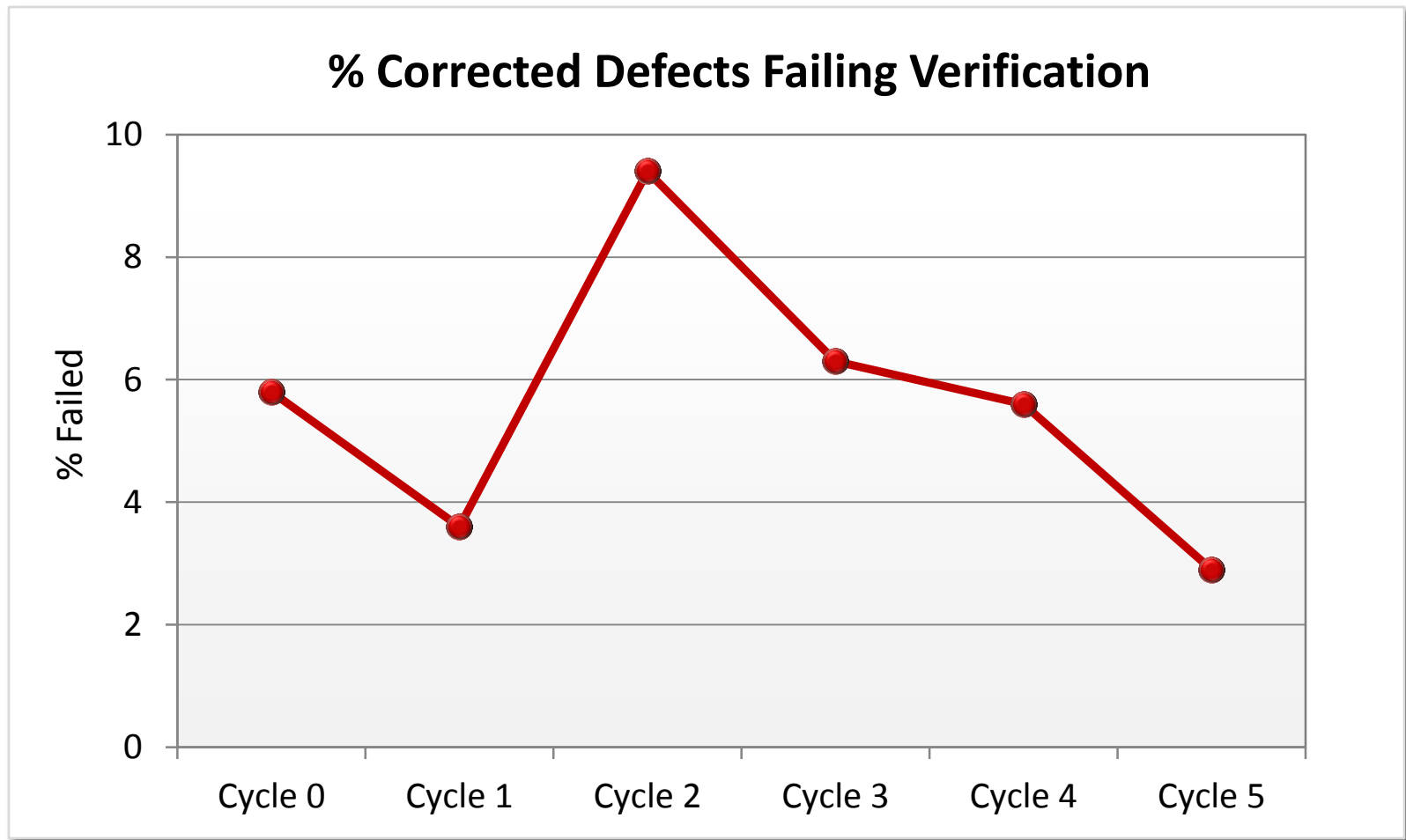
**Schedule Predictability: 16.2%**

# Results: *DxH 800 Reliability*





# Results: *DxH 800 Cleaner Code*



# Results: *DxH in the Field*



## DXH 800 2.0

Released:	6-29-2012
Installs:	Over 30 installed
Results to Date:	A few missed defects Minor update being planned Root cause identified Process will be updated

**"This is 98% better!"**  
- Our Customers

# Results: *Adoption by Engineers*

## Quotes from DxH / SMS Team

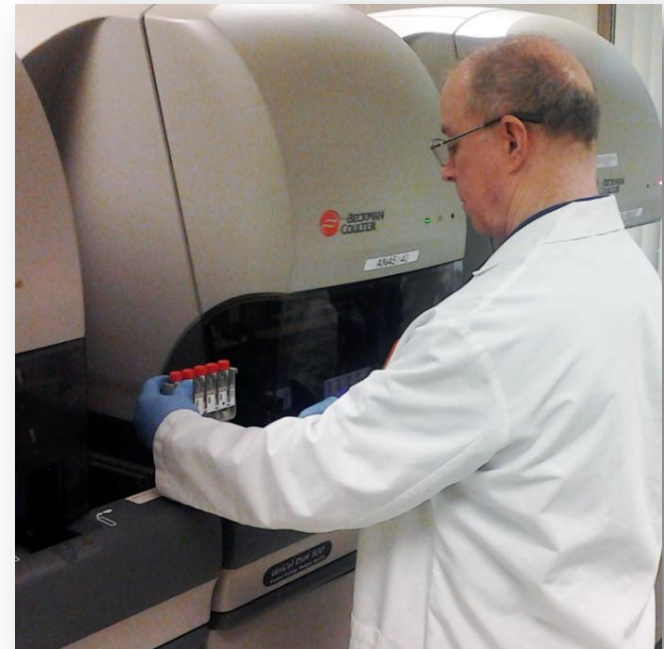
- “What's different now is that they used to ask us to fix 90 SCRs by a certain date, we'd try, and never make it. Now they ask us to do 90 SCRs and we let them know what we need to do it, or what we can do in the time frame - **we know all our rates.**” – *JC*
- “**It's not chaotic any more. Not thrashing.** We're getting more work done in less time than ever before.” – *Ana*
- “**TSP is magic!**” – *Steve*
- “**We'll never go back to the way we used to do reviews.** We never gave them enough time, they never found anything. Inspections are working great.” – *James*
- “**We feel under control now.** Don't ever want to go back to the way we did things before—we now know how to avoid the DxH scenario and all of those protections are in the SMS plan.” – *Andrea*



# Lessons Learned

**Possible to introduce TSP to a project late in development and still realize significant improvements.**

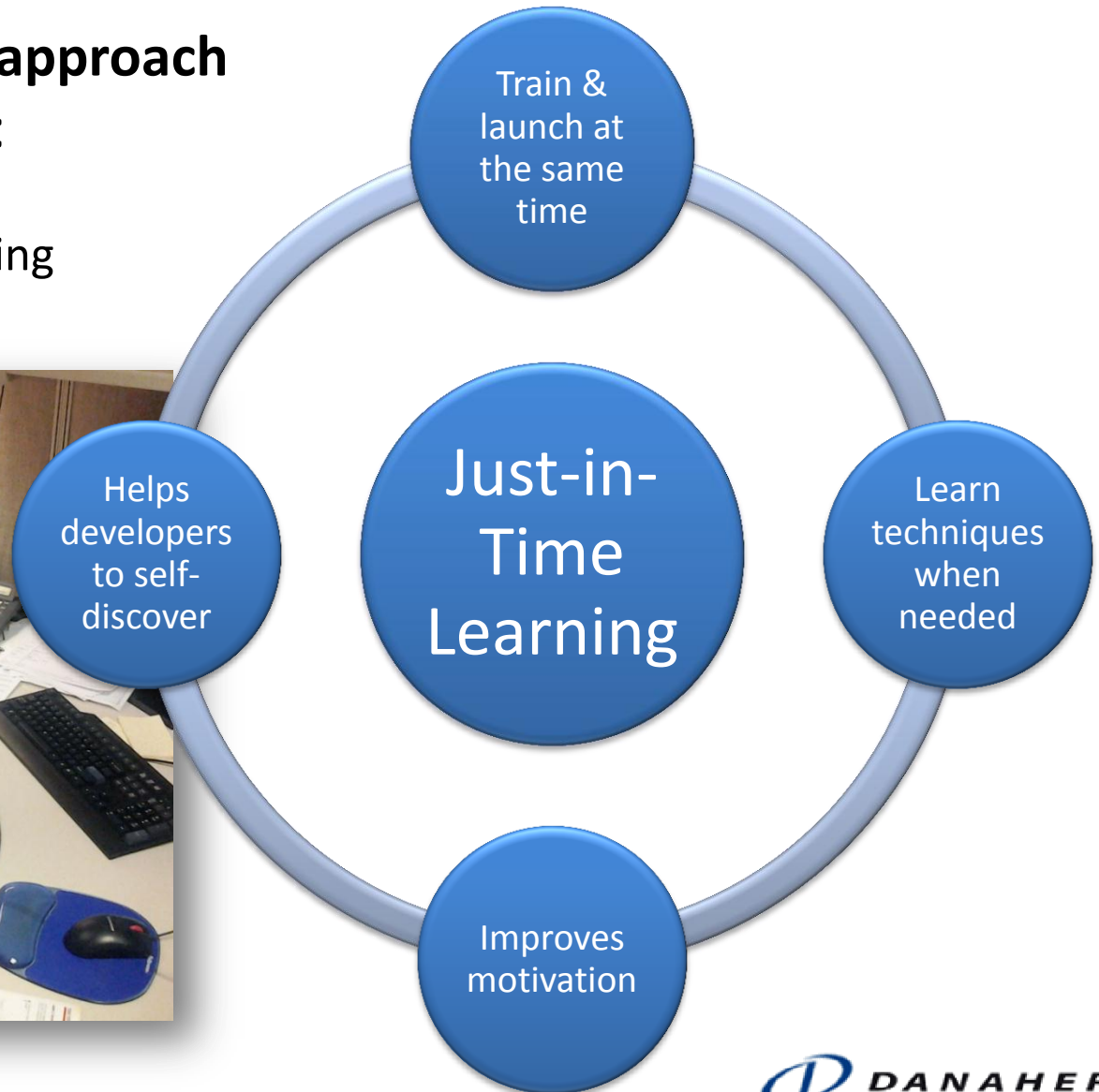
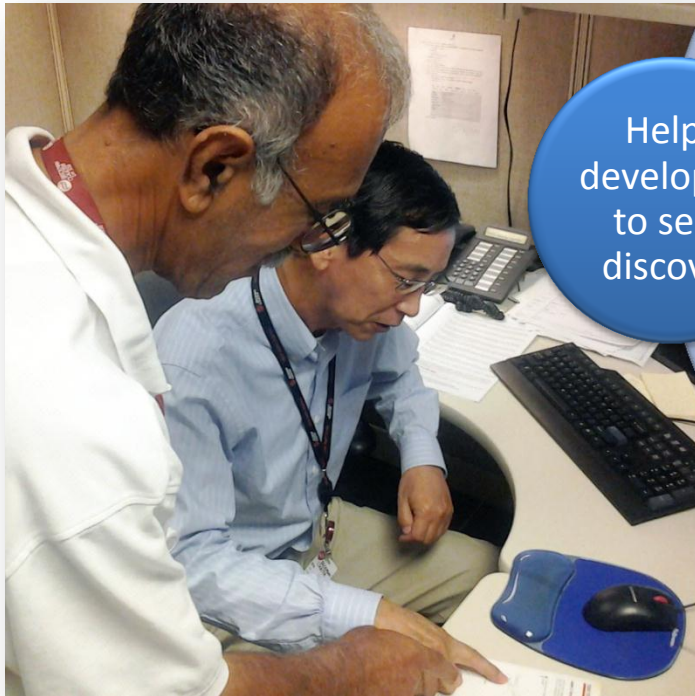
- Some TSP concepts are independent of where introduced
  - Personal reviews
  - Inspections, inspection rates, tracking
  - Planning and estimation
  - Knowing your rates/historical data
  - Use of task hours
  - Role managers/team structure
  - Coaching of team lead
  - Collection and use of data essential to improving



# Lessons Learned

## Just-in-Time Learning approach worked in this context

- Already in System Test
- Very little time for training



# Lessons Learned

## Coaching is essential to being effective!

However the training is initially delivered...

**learning, questions/answers, course corrections**

...happen throughout the project



Project management is not a **ballistics problem**.

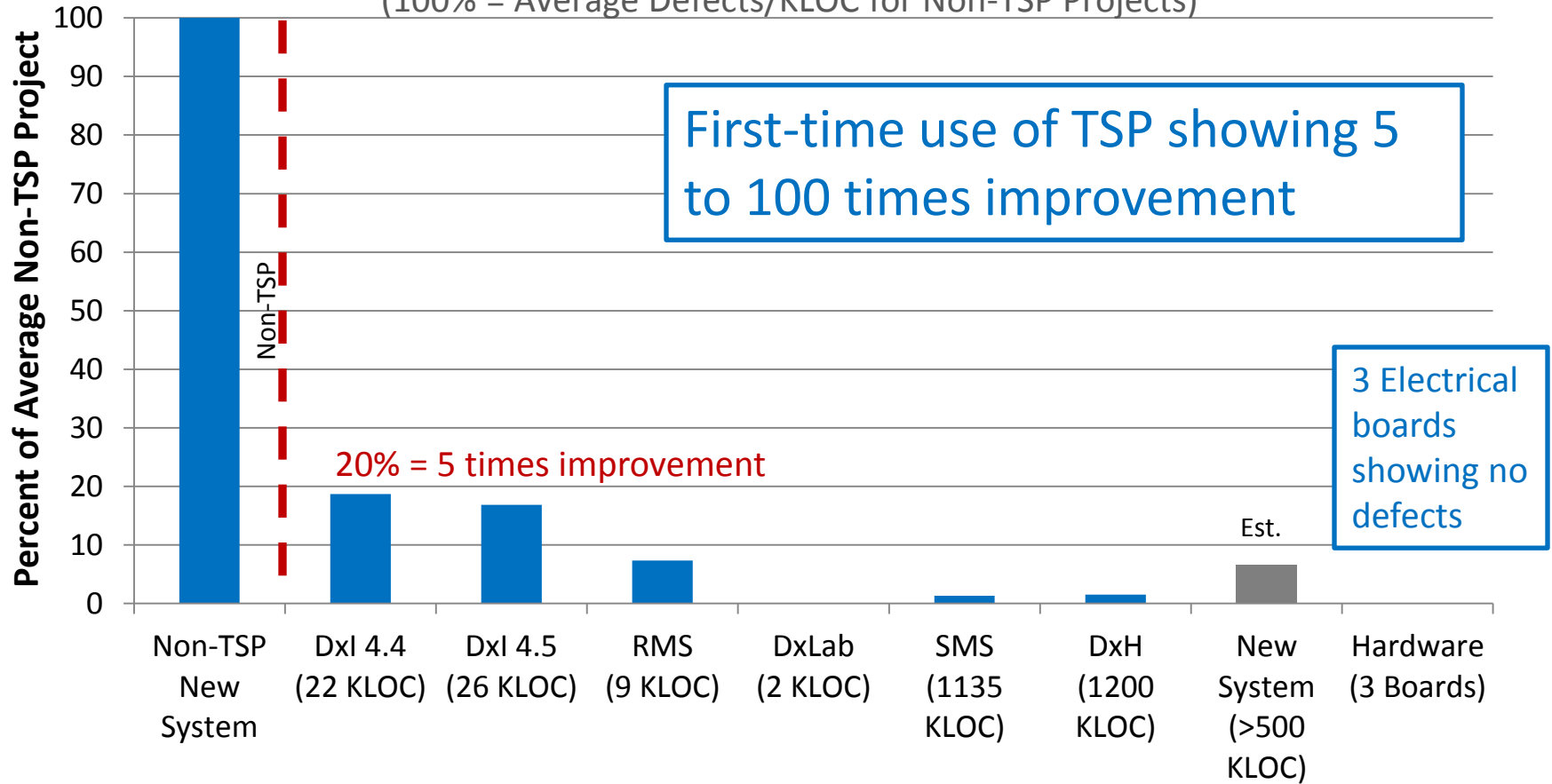
It acts more like a **guided missile**.



# Other Beckman Coulter TSP Results

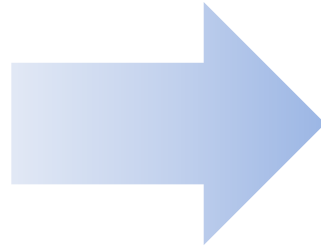
## Defects/KLOC as Percent of Non-TSP Average Project

(100% = Average Defects/KLOC for Non-TSP Projects)

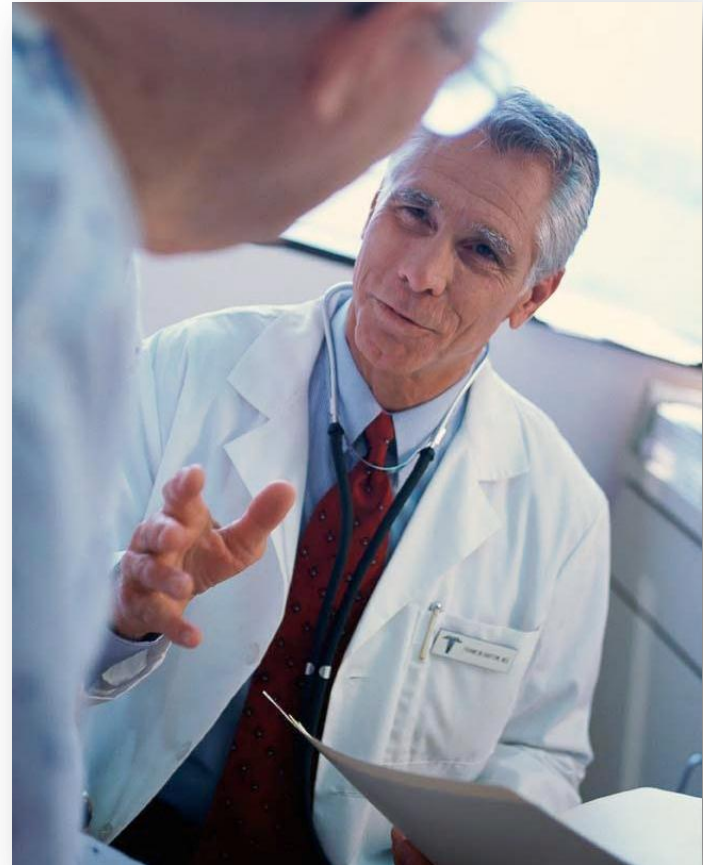




# Beckman Coulter: *Simplify . Automate . Innovate*



TSP



**Our Focus: Reliable Test Results**

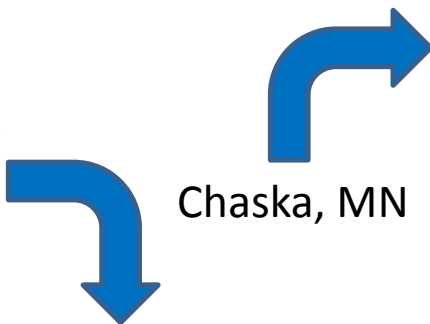
# Acknowledgements

- **Carl Wyrwa** (*CW Software Solutions, Inc.*) & **Lourdes Villanueva** (*Beckman Coulter*)
  - Uber coaches
- **Alan Willett** (*Oxseeker, Inc.*)
  - Helped develop our customized approach
- **Jim Over** (*SEI*)
  - Supporting our studies on Improvements and Sustainability
- **JC Urbina** (*Beckman Coulter*)
  - Team Lead of DxH
- **Andrea Britton** (*Beckman Coulter*)
  - Team Lead of SMS

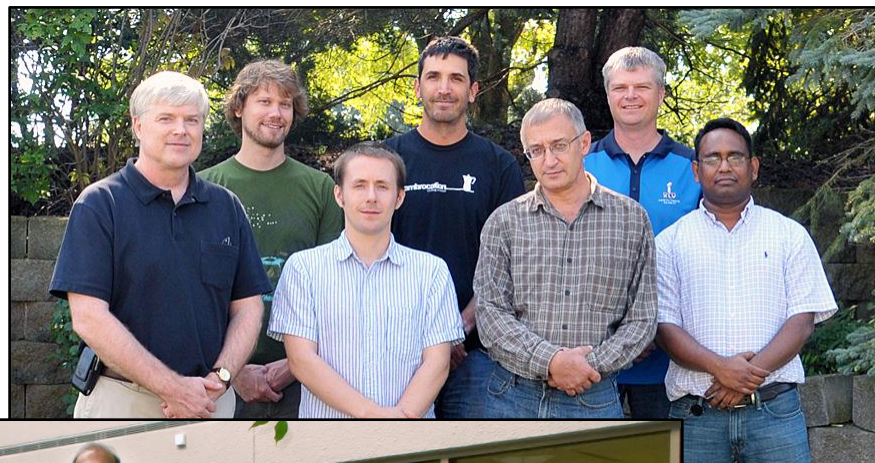
# Acknowledgements

## The Team!

Miami, FL



Chaska, MN



Brea, CA



# Questions?

Scott Van Eps: [svaneps@beckman.com](mailto:svaneps@beckman.com)

Rick Marshall: [rmarshall@beckman.com](mailto:rmarshall@beckman.com)