Practical Process Improvement

The Story of the
SP5 Testing Process Improvement Project

Chad Haggerty
Director, Quality
October 4 2007

Agenda

- Company Overview
- Process Improvement @ Ulticom
- SP5 Improvement Project
- Summary and Learnings

Company Overview

Ulticom Overview

The Company

- Leading provider of communication signaling components and systems enabling value-added services
- Current head count 270
- Sales, development, and support in North America, Europe, and Asia
- TL 9000 Certified

History

- 30-year history of innovating products and services for the communications market
- Public company since 2000

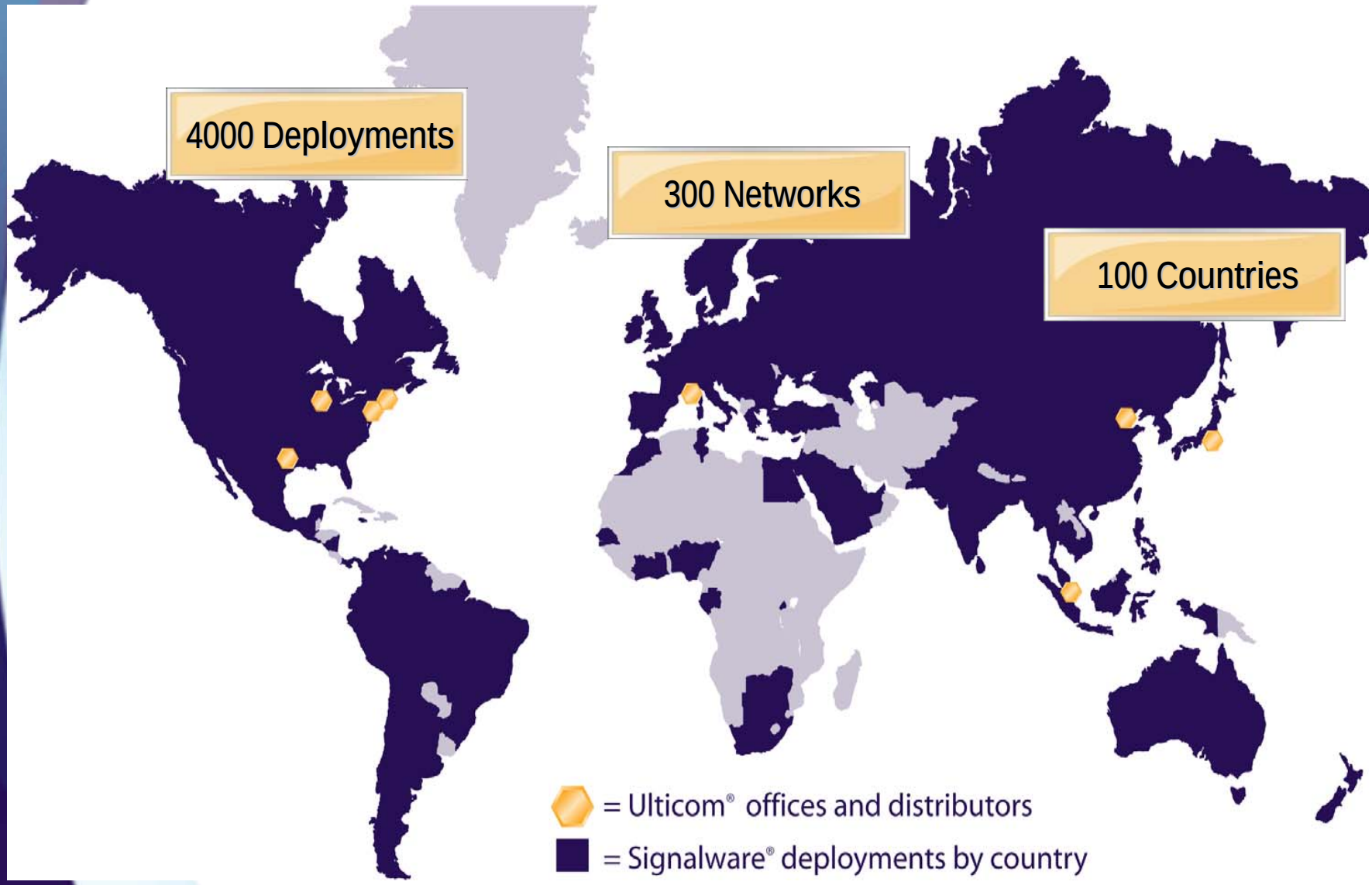
Markets

- Markets : Switching, Mobility, Payment, Messaging, and Location services in wireline and wireless networks
- Customers: Over 55 active system and service provider customers deployed in Over 4,000 deployments in 300 networks and 100 countries
- Growth driven by increasing subscribers and wireless service usage

Products

- Signalware – Multiprotocol signaling platform to develop and deploy applications
- *n*Signia – Signaling gateway to cost-effectively bridge SS7 and IP signaling networks
- Signalcare - high-value service and support programs to customers deliver on the next generation of communications services

Global Presence



Process Improvement @ Ulticom

Process Improvement @ Ulticom

FY07 Objective: Establish the data and methodology needed to drive coordinated improvement projects across the organization resulting in quantifiable improvements in quality or productivity

- Develop a process improvement process which incorporates
  - Six Sigma Principles
  - Lean Principles
  - Cost of Quality Model
  - Agile Principles
- Provide advanced quality training for Process Improvement
  - Hired Process-Fusion for training and consulting on Lean Six Sigma for small companies
  - Develop In-house training on data analysis
- Prove the value
  - SP5 Testing Project
- Quantify and Track Improvement Successes
  - Established Quality web-site and communication plans

Work Smarter NOT Harder!

User Name:
Password:
Login

HOME | APAC | CUSTOMER SUPPORT | EMEA | ENGINEERING | ES | HR | IT | LEGAL & FINANCE | QUALITY | SALES & MARKETING

Employee Search

Search

Quality Links

- Who We Are
- Quality Infrastructure
  - Process Library
  - DCR
  - PDCF
  - Process Owners
  - Training
- Quality Assurance
  - Reviews Database
- Process Compliance
  - TL 9000
  - Compliance
- Process Improvement
  - Agile
  - LEAN/Six Sigma

Process Improvement

"Work Smarter, Not Harder"

What is it? Constantly improve our product quality and productivity.
What should you do? Challenge the system and think "Outside the Box" for better ways. ASK WHY?
Quality Department's Charter Incorporating industry best practices (Agile, TQM, Cost of Quality, Lean Six Sigma, etc.)

FY07 PROCESS IMPROVEMENT $$$ SAVINGS

SP5 Testing Project

- *Web-ex Overview Presentation*
- *Summary of Results*

My Shortcuts

Note:
You must be logged in to view this portlet. Enter your Username and Password at the top of the page and click 'Login' to Personalize

Favorites

my.ulticom.com
www.ulticom.com

World Time

Mt. Laurel:	Tue, 10/2 15:40
Sophia:	Tue, 10/2 21:40
Singapore:	Wed, 10/3 03:40
UTC:	Tue, 10/2 19:40

Please click *HERE* for other cities or countries

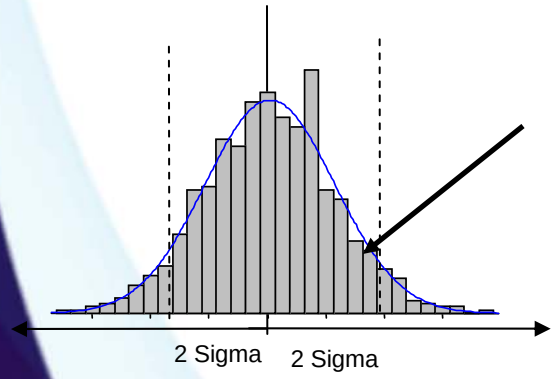
Agile

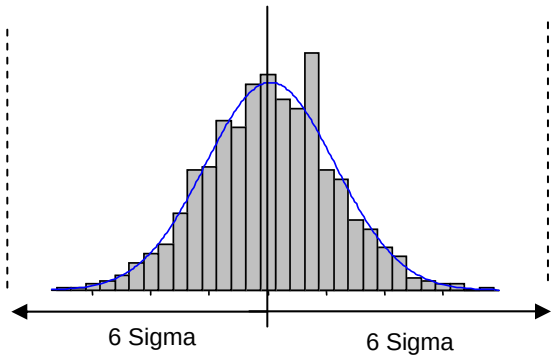
- Agile is a set of principles and values that promote a collaborative adaptable development process which frequently delivers value add products to customers.
  - Agile Practices and methods (Scrum, Extreme Programming, etc) are the means to an end to instill these principles and values into a company
- Agile is a company wide change
  - Customer involvement
  - Collaboration
  - Continuous testing and product delivery
- Agile is not a prescriptive process
  - Set of principles to be applied with common sense to Ulticom's environment
- Agile promotes team empowerment
  - Own the quality of the product
  - Own how they work (processes)
  - Own when and how much they deliver

Six Sigma

- A systematic approach to quantifiably solve problems and optimize critical business processes through both information and statistical analysis.
  - **Methodology to drive ROI driven improvements (DMAIC)**
  - **Tools to improve process performance**
- Successful adaptation of Six Sigma can lead to dramatically improved business performance, profitability, and substantial increases in overall customer satisfaction.

- Achieving a sigma level means that out of every 1,000,000 things you make, process, etc., the following number will be defective:
  - 2 Sigma = 308,538 defects
  - 3 Sigma = 66,807 defects
  - 4 Sigma = 6,210 defects
  - 5 Sigma = 233 defects
  - **6 Sigma = 3.4 defects**





Lean

- **Lean** is a generic process management philosophy derived mostly from the Toyota Production System (TPS)[1] but also from other sources. It is renowned for its focus on reduction of the original Toyota 'seven wastes' in order to improve overall customer value. Lean is often linked with Six Sigma because of that methodology's emphasis on reduction of process variation (or its converse smoothness). Toyota's steady growth from a small player to the most valuable and the biggest car company in the world has focused attention upon how it has achieved this, making "Lean" a hot topic in management science in the first decade of the 21st century.
  - **Production Capacity = Work + Waste (Muda)**
  - **Goal is complete elimination of waste**
  - Waste is defined as anything that does not directly add to the production of something that the customer will pay for
  - Seven Examples of waste
    - Defects
    - Overproduction
    - Inventory
    - Extra processing
    - Unnecessary motion
    - Transportation
    - Waiting

[1] Womack, James P., Jones, Daniel T., and Roos, Daniel (1991), The Machine That Changed the World

Source: Wikipedia

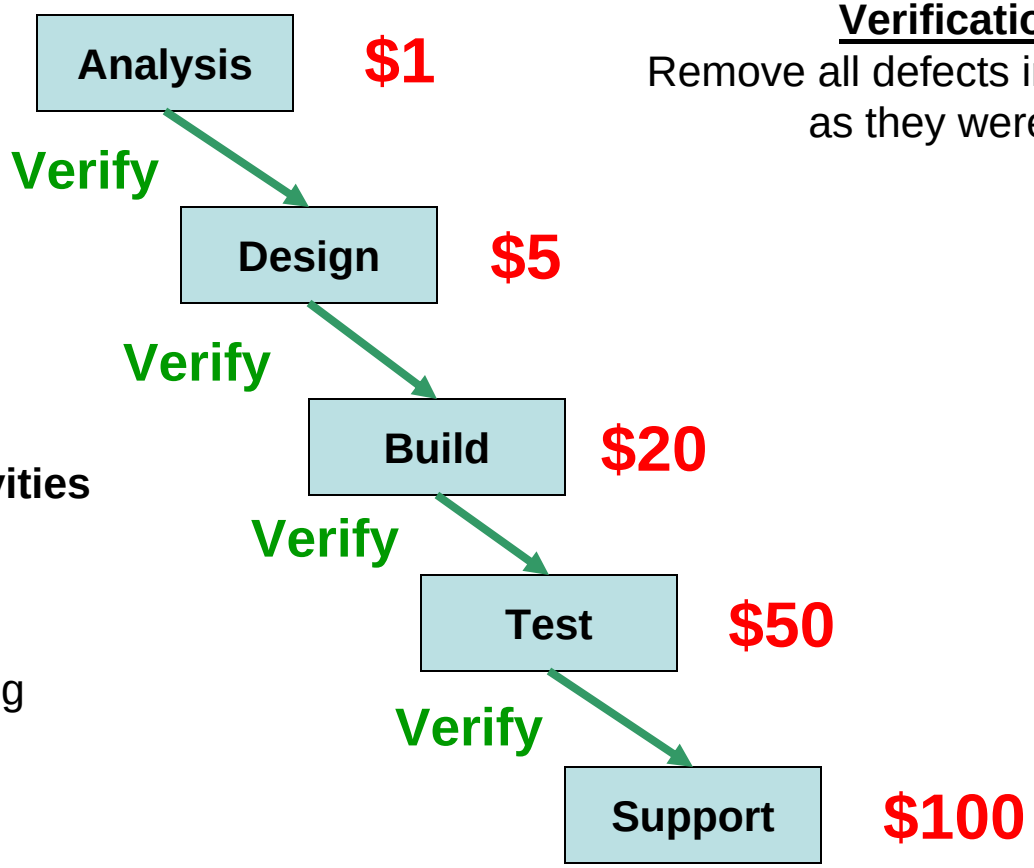
Cost of Quality

- Dealing with nonconformances (Internal and External).
  - Quality problems detected prior to product shipment.
  - Quality problems detected after product delivery.
- Appraising the level of quality.
  - Discovering the level of nonconformances.
  - Quality control gating.
- **Preventing poor quality from occurring**.
  - Efforts to define quality, set quality goals, standards, and thresholds. Quality trade-off analysis.
  - Efforts to prevent poor product quality or improve process quality.

An ounce of prevention can go a long, long way!

Source: **Using the Cost of Quality Approach for Software** by **Herb Krasner**

The later a defect is found in the process, the most costly it is to fix!



Verification Goal

Remove all defects in the same phase as they were injected

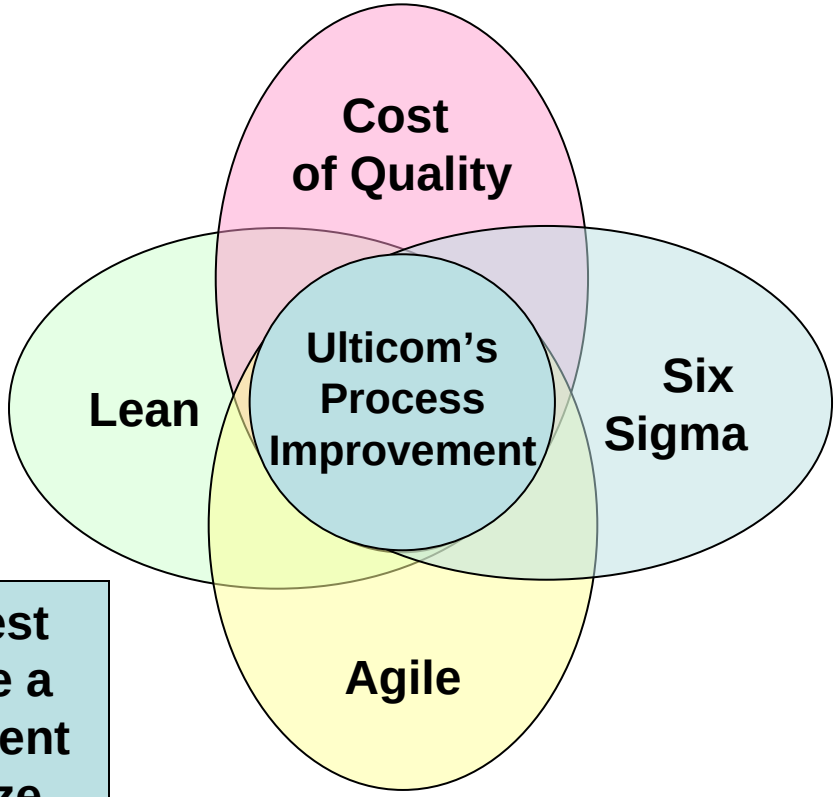
Verification Activities

- Artifact Reviews
- Unit Testing
- Component Test
- Integration Testing

Source: Boehm, B., Software Engineering Economics

Ulticom's Process Improvement

Measurement of Quality and Productivity



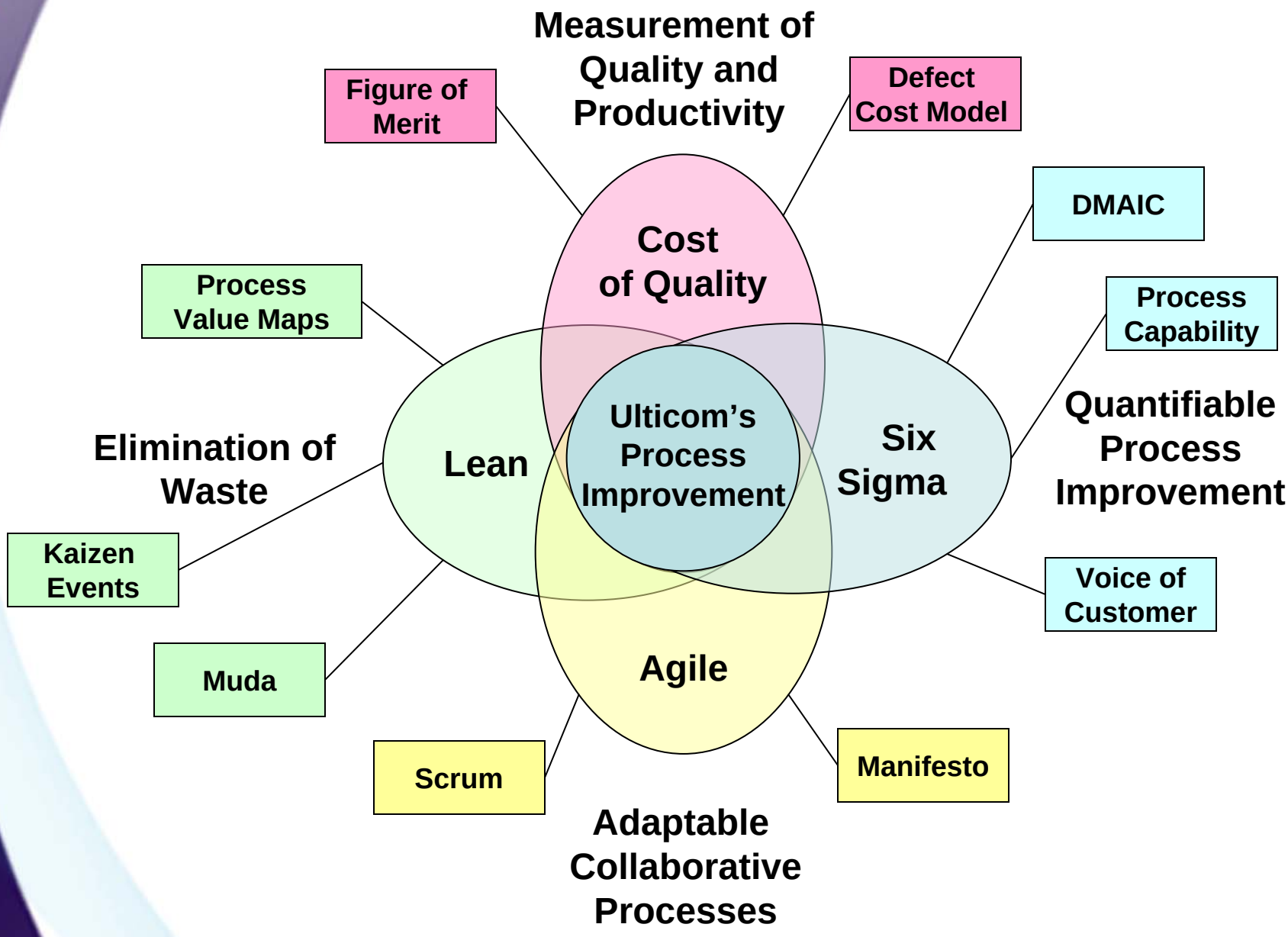
Elimination of Waste

Quantifiable Process Improvement

Utilize industry best practices to define a process improvement process for our size, environment, and culture

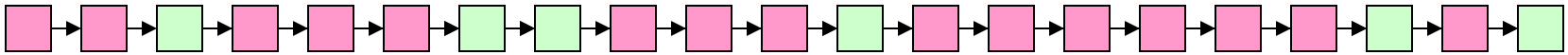
Adaptable Collaborative Processes

Ulticom's Process Improvement Tool Box

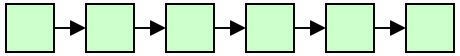


Lean and Six Sigma

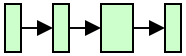
Improving a process…



Lean tools eliminate the waste….



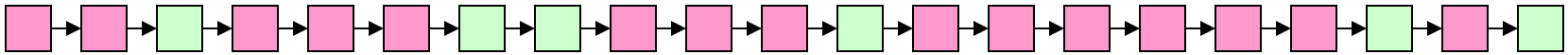
Six Sigma tools Improve the process….



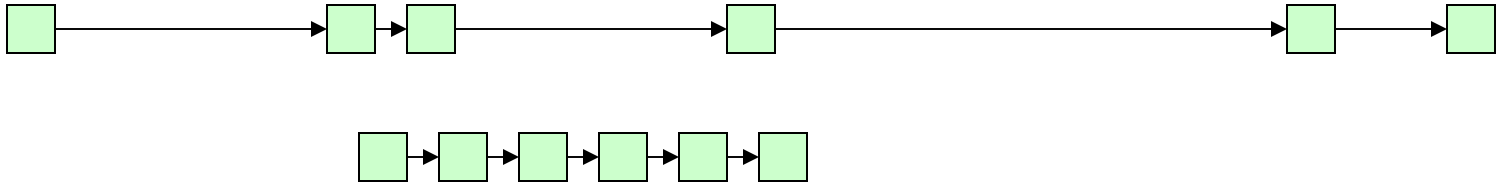
- Value Add - Waste

Process Improvement Program

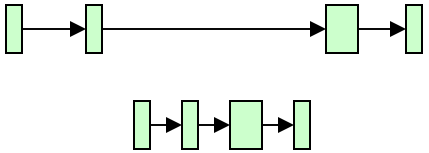
Improving a process…



Lean tools eliminate the waste….



Six Sigma tools Improve the process….



- **Lean**
  - Principles and tools to eliminate waste
- **Six Sigma**
  - Methodology to drive ROI driven improvements (DMAIC)
  - Tools to improve process performance
- **Cost of Quality**
  - Models and framework to capture re-work costs and impacts
- **Agile**
  - Principles for improvement efficiencies for product development **AND** Process Improvement

SP5 Improvement Project

Behind the scenes process improvement

The SP5 Improvement Project Story…

- Define Goals and Objectives
- Measure and Analyze the problem
- Determine and Implementation the solution
- Measure the results
- Opportunities going forward

S **Six Sigma Principle**

L **Lean Principle**

A **Agile Principle**

Q **Cost of Quality**

- **The Problem**
  - Signalware SP5 release
    - 15 different Operating Systems
    - 8 different protocols
    - 37 new features
    - Various defects fixes
  - Time and effort was becoming the critical factor jeopardizing the release date.

S

- **The Goal**
  - Dramatically reduce the SP5 testing effort by improving our test plans, processes, automated tests, and regression strategies, while maintaining or improving the product quality.
    - This will be accomplished by an initial Analysis and identification of improvements, and the use of Scrum to implement the improvements.

Measure and Analyze

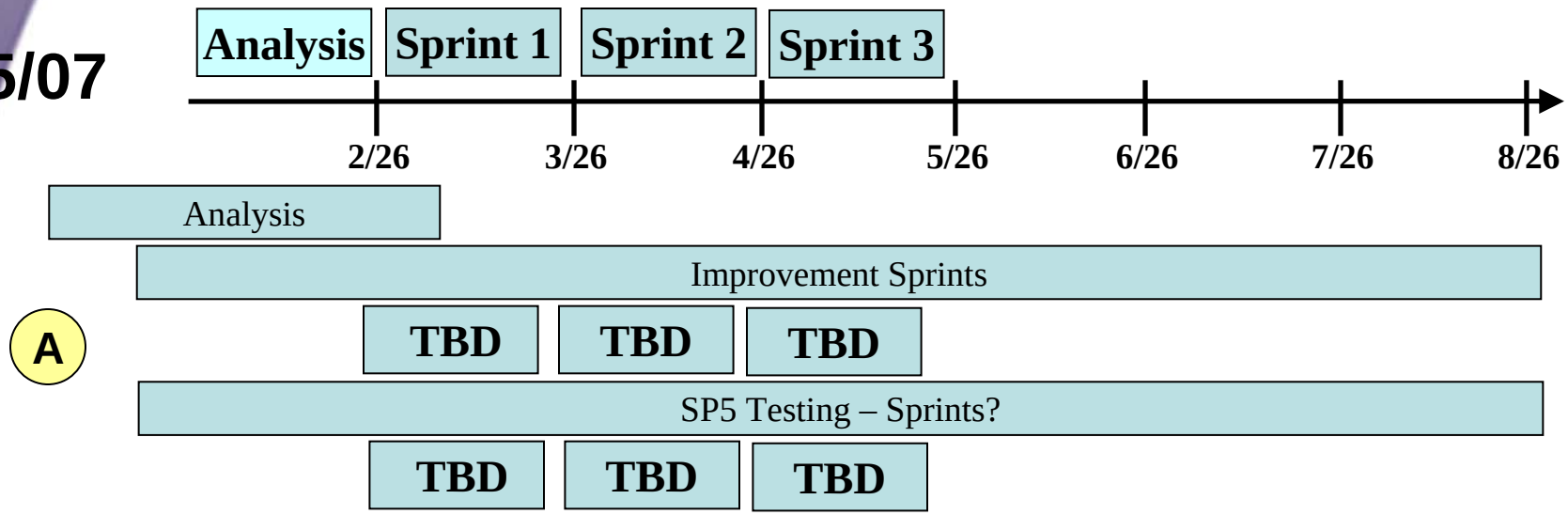
1. Determine the test case(s) that require the highest amount of efforts in order to select and prioritize candidates for automation.
2. Determine a general feel for the effort of all the testing.
3. Analyze current CR list broken down by category and severity – what does our current regression testing cover?
4. Define the criteria for when we need to specifically test a CR
5. Analyze current test cases, what is the overlap? (Regression vs. Functional)
6. Analyze different protocol testing to determine redundancies
7. Review OS specific test cases to determine overlap
8. Identify hardware resources needed\available
9. How will we capture\document analysis
10. Risk assessment of test cases to determine which one generates the most bugs and run them first.
11. Determine which test cases can be automated
12. Brainstorm ideas for making the test/bug fix process more efficient

A SP5 Improvement Team

- Product Owner: System Test Manager
- Scrum Master: Project Manager
- Scrum Team:
  - 5 Developers
  - 5 Testers
  - 2 Development Manager
  - 1 Quality and Process

High level Implementation Timeline

2/15/07



Analysis – Test Cases

A

L

- Test Areas were reviewed by developer/tester pairs to identify redundancy and opportunities for Automation
- Test Data was consolidated to a single Excel file and the data was sliced and diced using Macros and Pivot tables.
  - By OS, by Protocol, By Test type, By Effort, By Automation

S

Test	(All)
Status	(All)
Remove	(All)

Sum of Total Points		Op																		
Test Type	Test Area	MV	MV3	MV4	Red Hat	RHAS	RHAS3	RHAS4	RHAS5	SC 3.1 Upd	SC 3.1 Upd	SC 3.2	SLES	Sol10_OPT	Solaris 10	Solaris 9	Solaris10	Solaris9	(blank)	Grand Total
Feature	FR316						75											120		195
	FR416																	19		19
	FR469		365	205		36		460									115	464	19	1664
	FR476							180									80	350	20	630
	FR503						60											615		675
			100				80										140	240	20	580
			320				320						320				320	320		1600
	FR539		20	5		10	5	20									45	155	15	275
	R543		85				85						85				85	85	10	435
	FR552		16				6	5										16		43
	FR580						27	10									20	322	14	393
	R598				403					253	294	78			400	403				1831
	FR689			15			28	26									59	136	12	276
	FR765		151					217												368
	FR791																	130		130
Performance	P-Board Throughput Tests																		280	280
	P-Dual CE Tests							220									120	180		520
	P-Endurance Tests - PC0200							120									60	60		240
	P-Endurance Tests - PH0301						60	120										60		240
	P-Four CE Tests							240										320		560
	P-Netra V440 Layer Tests																	40		40
	P-Single CE Layer Tests							110										110		220
Regression	R-Extreme Ops Regression		216	548			772	728					488	1928			1166	1246		7092
	R-Install/OLU Regression		6	10			10	70					6				44	32		178
	R-Miscellaneous Regression		40				30										80	140		290
	R-Platform Regression		254				258	247	1				25				266	1551		2602
	R-RNC Regression			489																489
	R-Sigtran Regression			276			514	700					20				689	1103		3302
	R-SS7 Regression	39	200	22		106	503	970					1				226	2140		4207
Grand Total		39	1773	1570	403	152	2833	4443	1	253	294	78	945	1928	400	403	3515	9954	390	29374

~20% Redundancy Identified

Analysis – Defects

- Every defect is explicitly tested
  - Defect verified with Baseline, Fix Installed, Defect Fix verified
- L
- Do we need to explicitly test every defect?
  - What defect need explicit testing?
  - What defects can be verified by a current regression test?
  - What defects can be verified by unit testing?
- Split the defects into two categories
  - Cat 1 – All Criticals, All Majors, and Minors found by the Customer
  - Cat 2 – Minors found internally.
- Validate our data is accurate
- S
  - Used random sampling to validate our severity and type defects
  - Looked at re-work of minor defects to determine defect fix failure rate

Analysis – Defects

A

- All Cat 1 defects were reviewed by developer/tester pairs to identify the defects that could be verified by a current regression test.
- All Cat 2 defects were verified via unit testing and a formal sign-off by the developers who fixed the defect.
- In the example illustrates how we determined the effort savings
  - 14 Cat 1 defects were verified during regression testing
  - 47 Cat 2 defects were verified during unit testing
  - Saving the effort of testing 61 defect explicitly

			Testing		
CAT	Defect Type	Count	Explicit	Regression	Unit
1	Critical	2	1	1	0
1	Major	34	23	11	0
1	Minor with Sr	17	15	2	0
2	Minor with No SR	47	0	0	47
	Total	100	39	14	47

Analysis - Plan

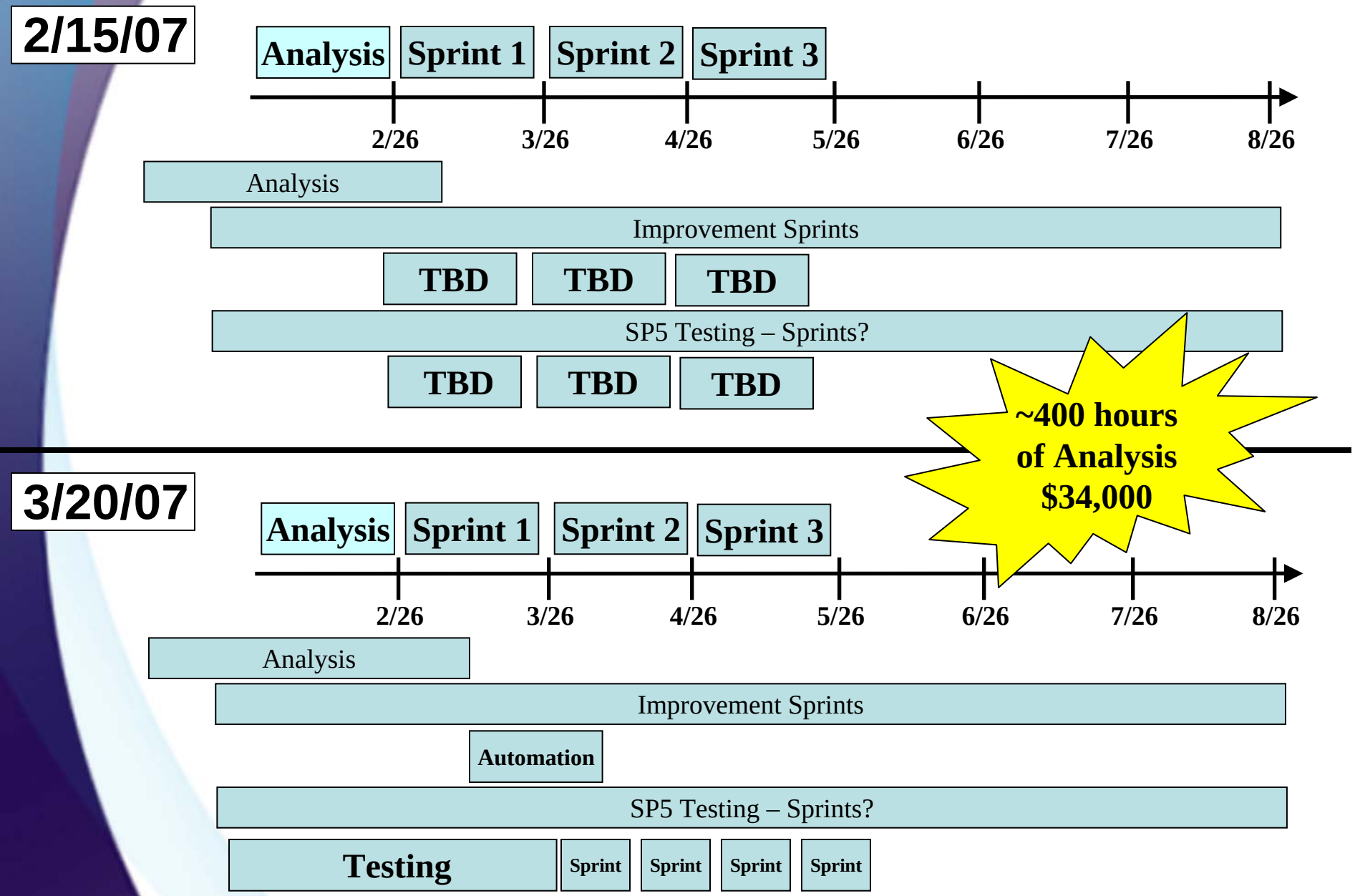
- War Room Established
  - Meeting and collaboration
- Daily Stand-up Meetings were held
  - Review progress and resolve issues

1. Hardware Team 1 (9:00)
2. Performance Team (9:15)
3. Feature Team 1 (9:30)
4. Install and Upgrade Team (9:45)
5. Feature Team 2(10:00)
6. Test Automation Team(10:15)
7. Scrum of Scrums (10:30)

Analysis – Build/Fix Strategy

- Formal builds occurred every 2 to 4 weeks
  - Defect Fix process cycle time is 4 to 6 weeks
    - Defect Opened, Approved, Fixed, Formally Built, Retested
- L How do we reduce the cycle time and complete testing backlog items (finish what we started)?
- A Collaborative Testing teams (Testers and Developers)
- More frequent builds
  - Formal Builds
    - Weekly builds (up to 16 OS builds)
      - Master schedule will determine the OS by week.
  - Informal Builds
    - Daily builds (on demand as needed)
      - Teams request build to correct defects and retest.

A Analysis - Timeline



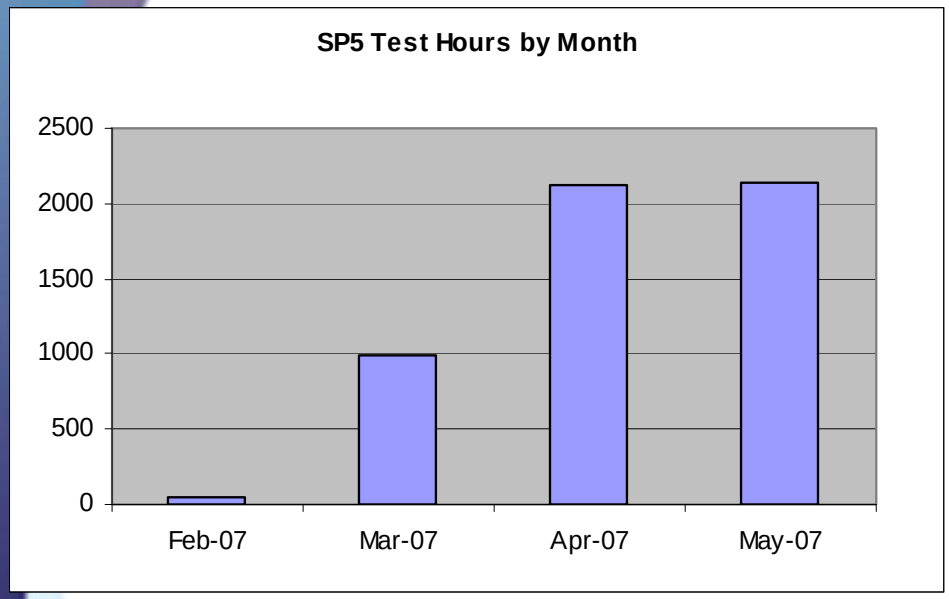
Implemented Solutions

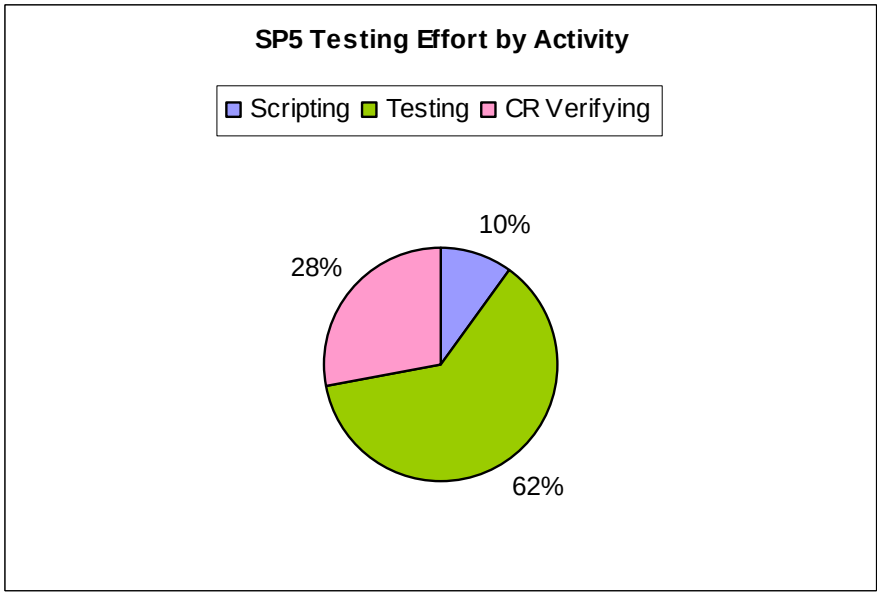
1. Eliminated 18% test cases
   - 20% reduction in testing effort.
2. Identified test cases to be automated
3. Eliminated explicit testing of 25% CRs
   - Analysis and development of a CR test strategy (do not test internal minor CRs)
   - Identification of CR that can be verified by regression testing
4. Development of a stream lined CR approval process with changes to our defect tracking tool
5. Development of a stream lined build process.
   - Weekly formal builds and on-demand informal builds
6. Implementation of Agile/Scrum principles
   - 5 Testing Scrum Team.

L S Measure the results…

1. Reduction of Testing Effort
   - Test cases and CR verification
2. Process Efficiencies (Productivity)
   - Stream lined CR and Build Process
   - Collaborative focused Scrum Teams
3. Cost of Delay Savings
   - What if it was delayed?

SP5 Testing Performance – Test Case and CR Reduction





Based on April and May Data					
Activity	Effort (Hrs)	Test points/ Defects	Hours Per	Eliminated Points/Defects	Hours Saved
Scripting	429	13265	0.03	5781	186.8
CR Verifying	1190	389	3.06	256	783.4
Test Execution	2641	13265	0.20	5781	1151.1
				Savings (Hrs)	**2121.4**
Loaded HourlyRate		$85.00		Savings ($)	**$180,317.51**

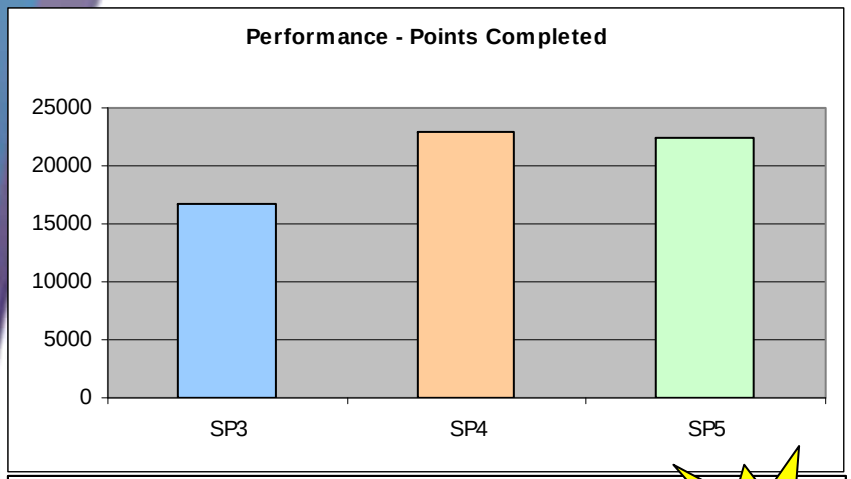
S Measure the results…

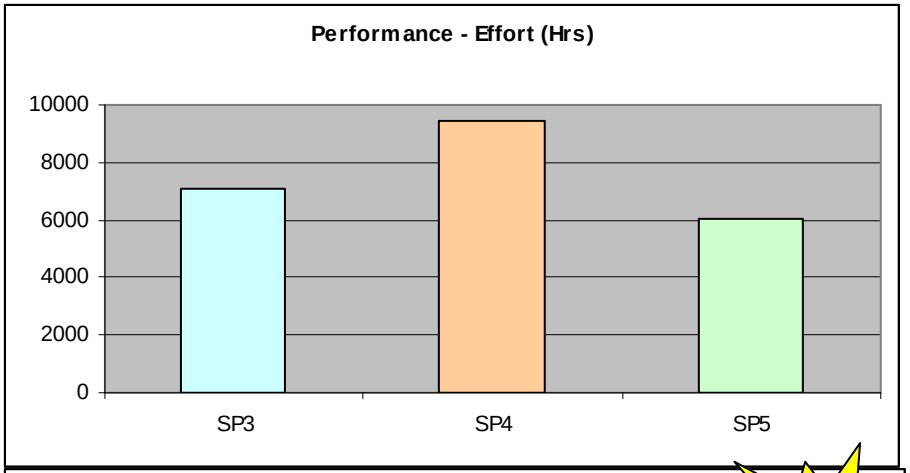
1. Reduction of Testing Effort
   - Test cases and CR verification
2. Process Efficiencies (Productivity)
   - Stream lined CR and Build Process
   - Collaborative Focused Scrum Teams
3. Cost of Delay Savings
   - SP5 Released on 6/11/07

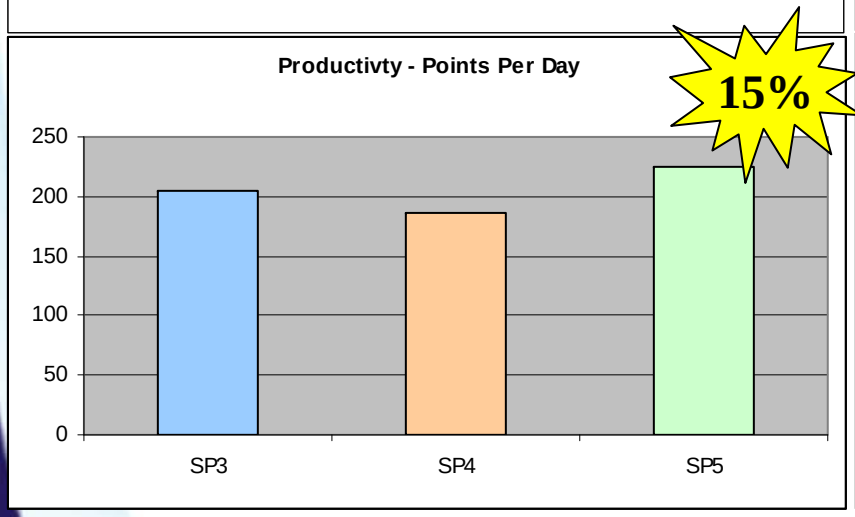
→ **$180,317**

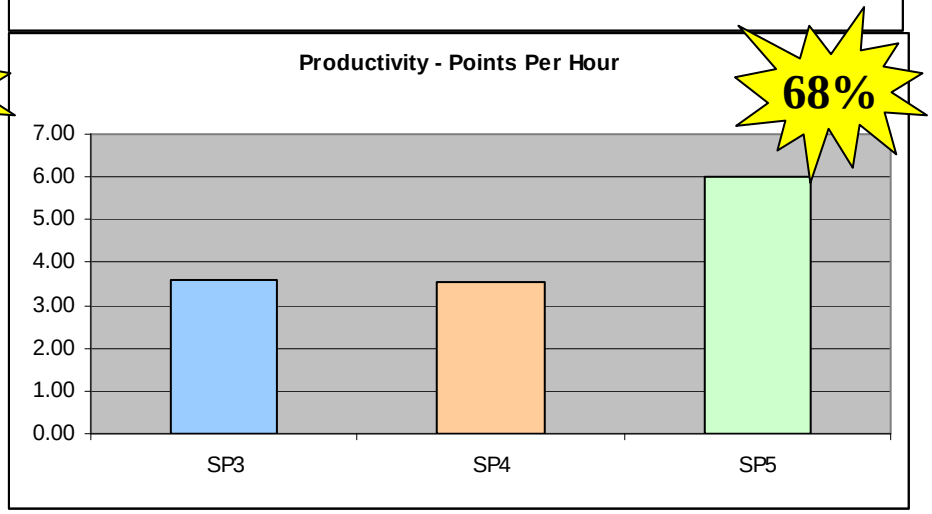
4 week delay

SP Test Performance Comparisons









Release	Start Date	End Date	Days	Points Completed	Point Per Day	Effort (Hrs)	Percent Scripting	Percent Testing	Percent CR Verifying	Testing Effort	Point Per Hour
SP3	Apr-05	Jul-05	81	16650	205	7092.8	16.53%	65.59%	17.87%	4652.5	3.57875
SP4	Aug-05	Dec-05	123	23017	187	9425.55	5.96%	68.43%	25.62%	6449.5	3.56878
SP5	Feb-07	May-07	100	22470	**225**	6019.25	10.1%	62.0%	27.9%	3731.7	**6.02139**

S Measure the results…

1. Reduction of Testing Effort → **$180,317**
   - Test cases and CR verification → **4 week delay**
2. Process Efficiencies (Productivity) → **15% - 68 %**
   - Stream lined CR and Build Process → **2 – 5 week delay**
   - Collaborative Focused Scrum Teams → **$90,158 - $225,397**
3. Cost of Delay Savings → **6 – 9 week delay**
   - SP5 Released on 6/11/07

Cost of Delay

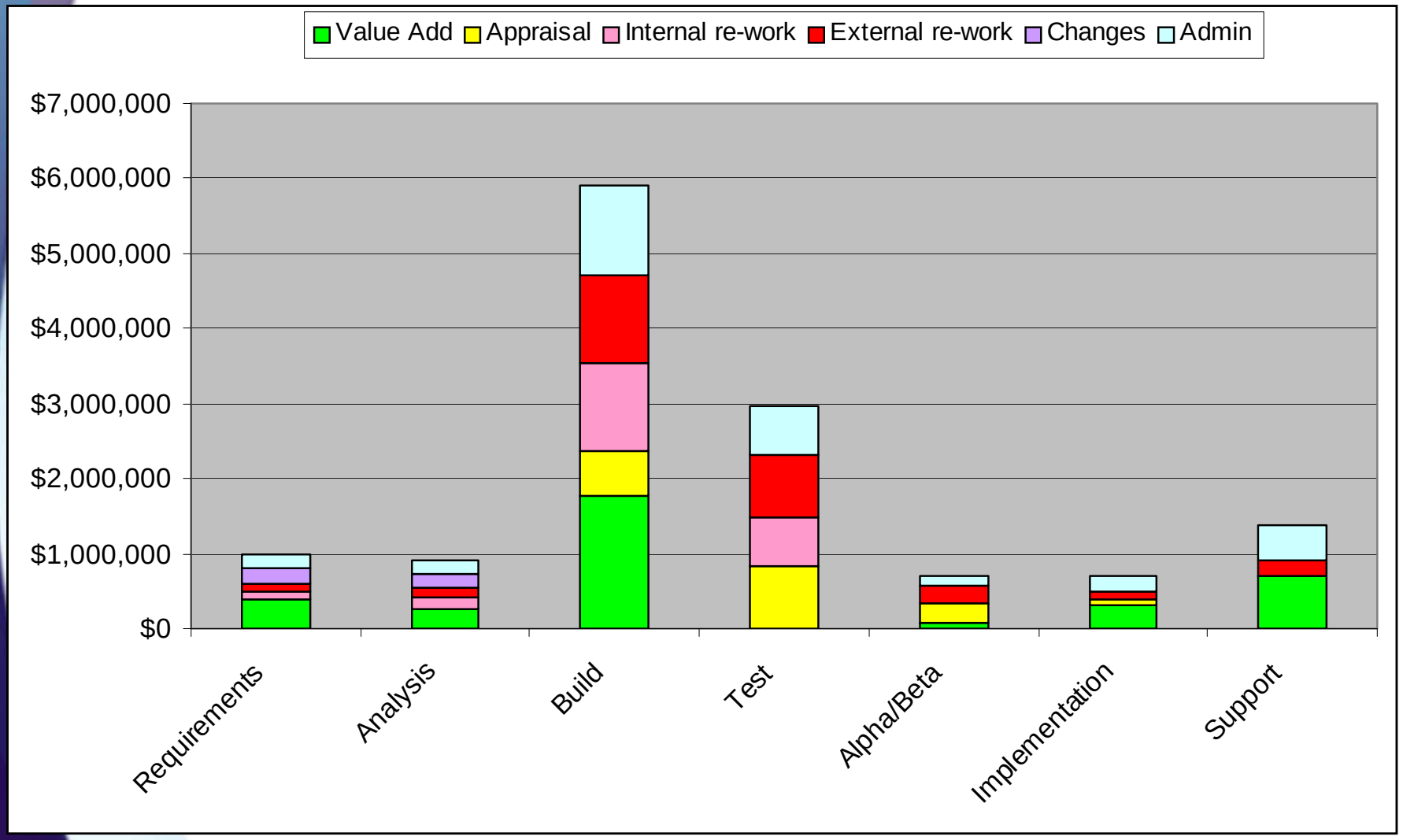
- What if SP5 was delay 6-9 weeks
  - What are the revenue impacts to SP5?
  - What are the cost and revenue impacts to our SP5 customers?
  - What are the revenue impacts to all other projects that would have been delayed?
  - What other impacts…..
  - $$$$

S Measure the results…

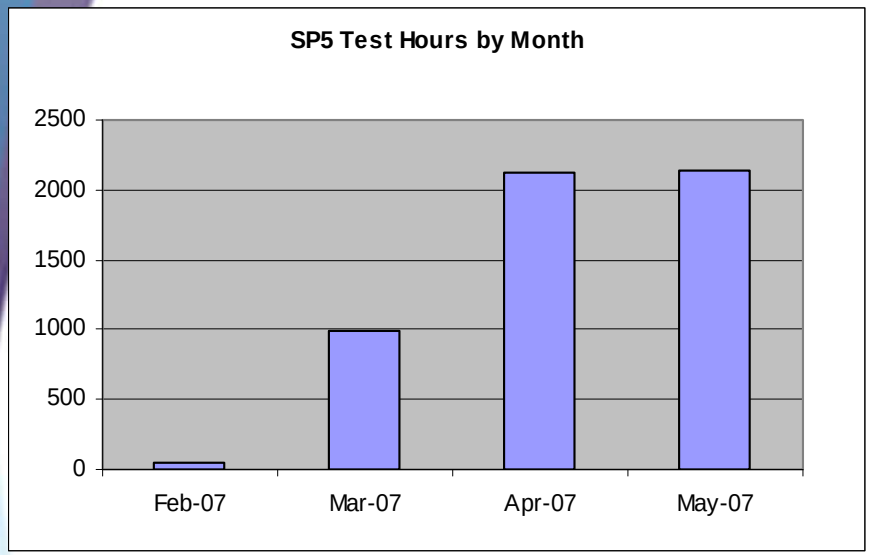
1. Reduction of Testing Effort → **$180,317**, **4 week delay**
   - Test cases and CR verification
2. Process Efficiencies (Productivity) → **15% - 68 %**, **2 – 5 week delay**, **$90,158 - $225,397**
   - Stream lined CR and Build Process
   - Collaborative Focused Scrum Teams
3. Cost of Delay Savings → **6 – 9 week delay**, **$$$**
   - SP5 Released on 6/11/07

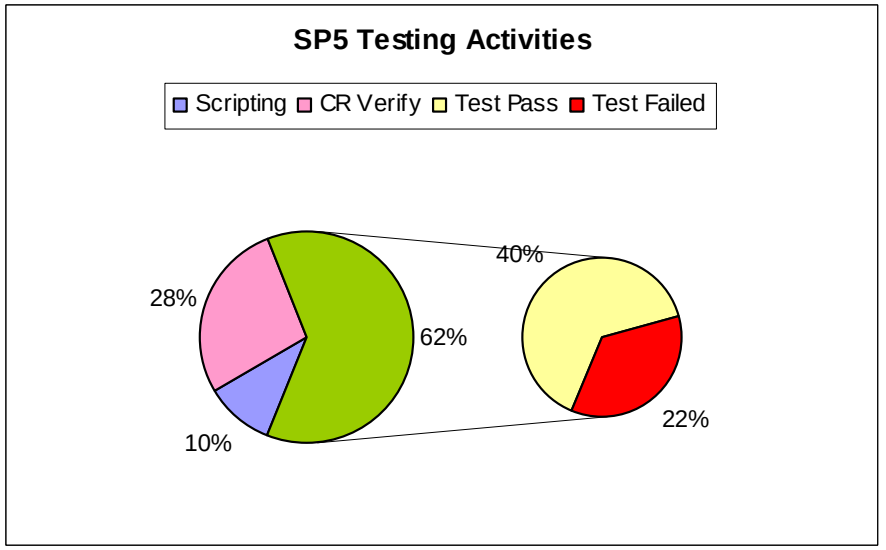
	Low	High
Improvement Costs	$34,000	
Improvement Savings (minimum)	$270,476	$405,714
Net Savings	$236,476	$371,714
ROI	8:1	11:1

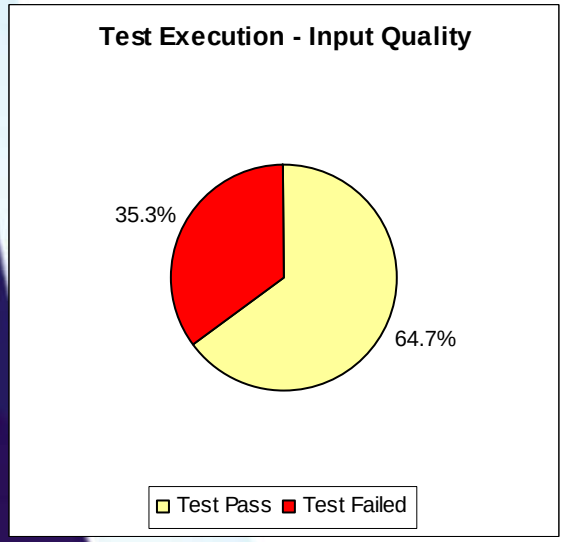
Cost Of Quality

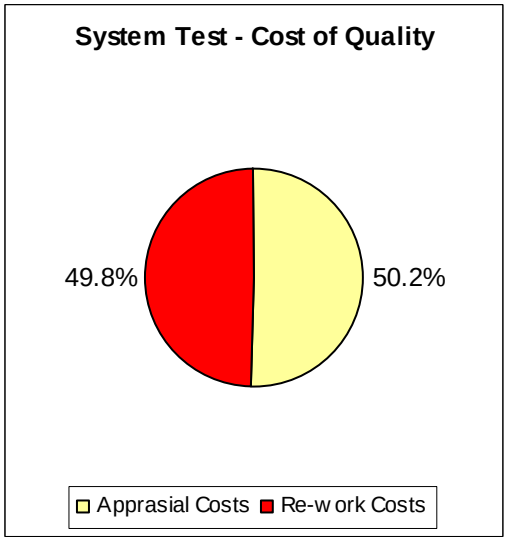


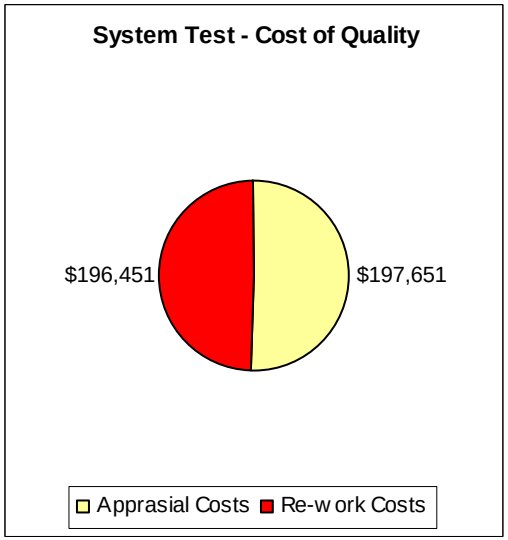
Opportunities – Cost of Quality – Internal Re-work











Summary and Learnings

Summary

- A successful collaborative improvement project with quantifiable savings
  - 236,476 to $371,714 with a 8:1 to 11:1 ROI
  - Repeatable Savings for Future Releases
  - A 6-9 week project delay avoided
- Use of Six Sigma, Agile and Cost of Quality principles and tools
- Identification of future opportunities for improvement.