Why CMMI Isn't Enough?

Systems Engineering Conference
October 20-23, 2008

Anita Carleton
Tim Chick

Performance Results of CMMI – Based Process Improvement

Performance Category	Median Improvement
Cost	34%
Schedule	50%
Productivity	61%
Quality	48%
Customer Satisfaction	14%
Return on Investment	4:1

WARNING: Results may vary!

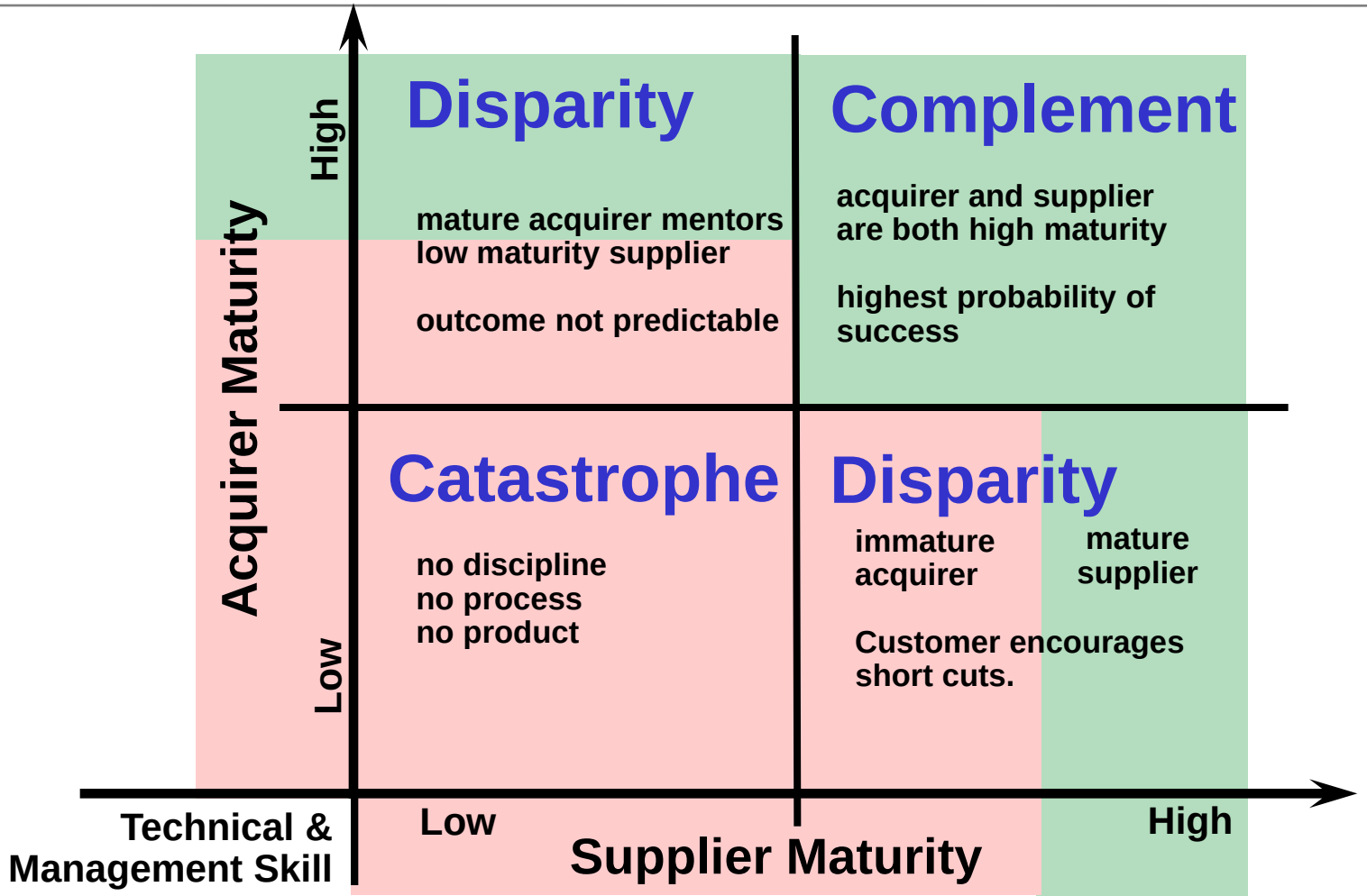
Source: http://www.sei.cmu.edu/pub/documents/06.reports/pdf/06tr004.pdf

Why results vary - 1

Two different approaches to CMMI based Process Improvement:

- Bureaucratic improvement that comes to life only when assessments are to be performed
- Improvement efforts that are based on achieving business objectives which are embedded into the culture of an organization and actively supported by the entire staff

Why results vary - 2

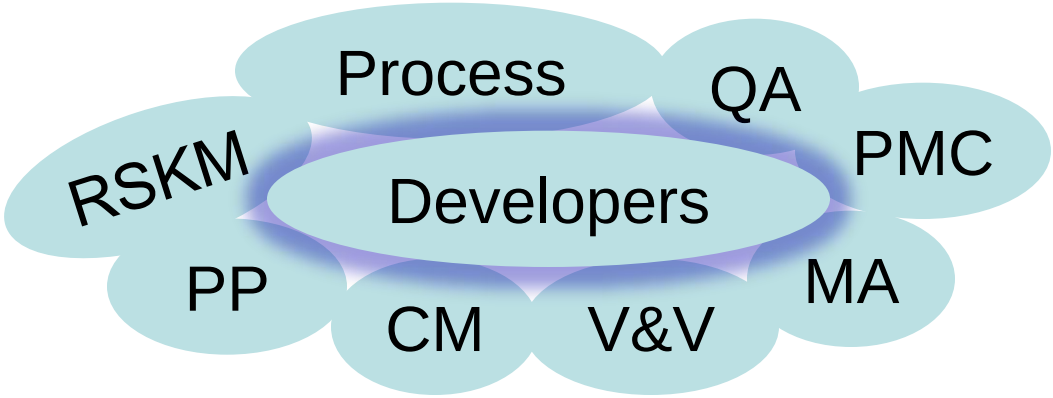


Source: http://www.dau.mil/pubs/dam/11_12_2006/11_12_2006_chi_nd06.pdf

Bureaucratic Improvement

Bureaucratic Improvements can be very successful in changing the organizational culture. However it doesn't fundamentally change the developers individual behavior or processes.

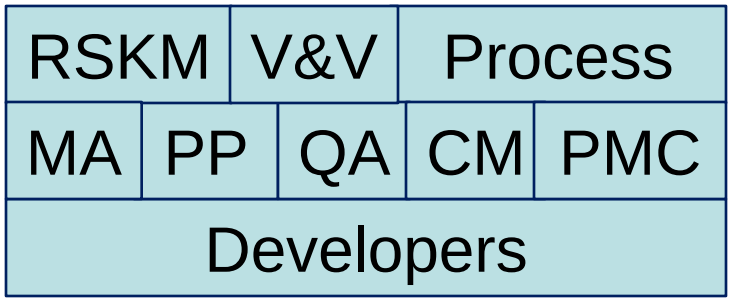
Resulting in continued quality, cost and schedule issues. Because ultimately only the developers can control the quality of the product, which directly impacts the cost and schedule.



How to get the performance you expect using CMMI

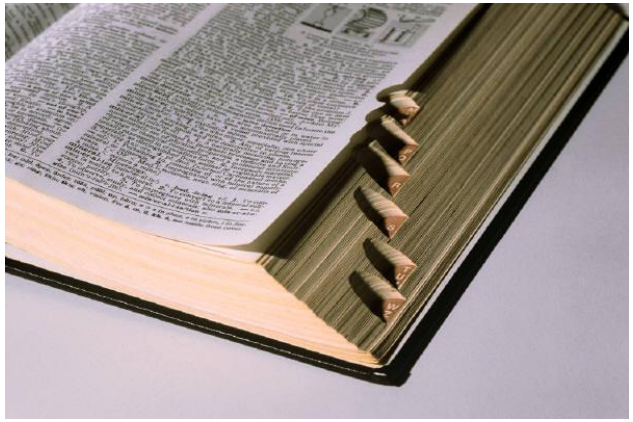
Improvement efforts that are based on achieving business objectives which are embedded into the culture of an organization and actively supported by the entire staff:

- Achieving a maturity rating doesn't guarantee improved performance
- To get high performance, you need to build a solid foundation from the beginning
- Performance becomes an enabler for high maturity



Definitions

High Performance –
High performance means obtaining superior outcomes.



High Maturity –
Implementing the concepts and practices at levels 4 and 5 of CMMI.

High Maturity Practices –
The "specific practices" and "generic practices" at levels 4 and 5 of CMMI.

Align Business Objectives

Are we getting more business moving to a higher maturity?

Are we shipping (releasing) higher quality products?

Do we have better performance?

Do our products have more functionality?

Are we reducing our costs?

Are we meeting our schedules?

How do we get high performance from high maturity?

Prerequisites for High Performance

Before an organization can perform high maturity activities, it must:

- Gather and use data at all organizational levels
- Defined operational processes that specify how and when the data are gathered
- Faithfully execute the defined processes

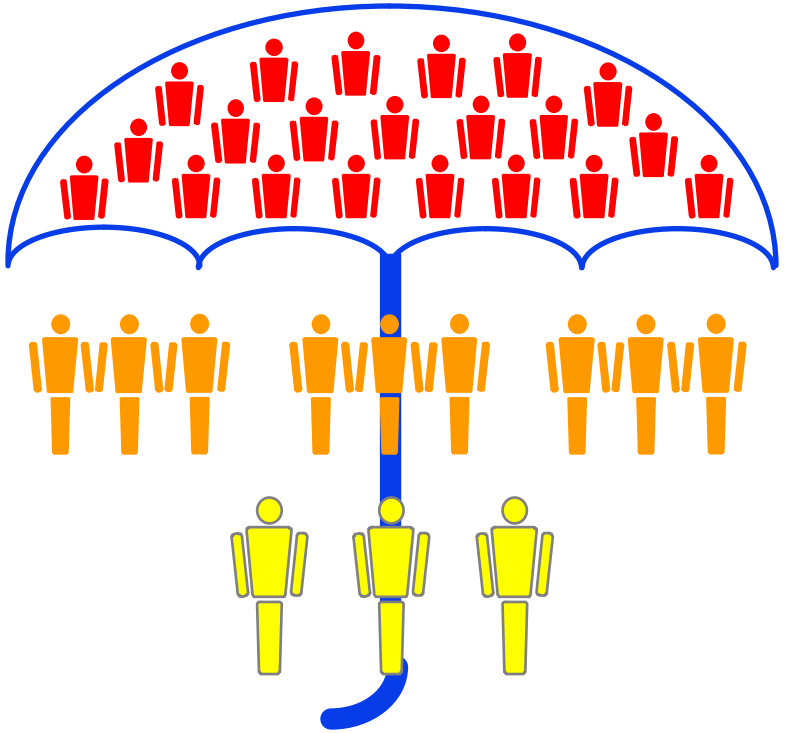
This implies that individuals and teams gather data on their own and use the data to plan and perform work

To Get High Performance, Address Team and Individual Discipline

A high-performing ***organization*** must be built of high performing teams.

High performing ***teams*** must be built of high-performing individuals.

High-performing ***individuals*** must be disciplined to gather and use their own data.



For a successful case study showing the integration of CMMI and TSP, please see “CMMI Level 5 and the Team Software Process” by Webb, Miluk, and Van Buren in CrossTalk April 2007.

http://www.stsc.hill.af.mil/crosstalk/2007/04/0704WebbMilukVanBuren.html

Operationalizing CMMI Practices

What does operationalize mean?

- To put something to use

What are characteristics of an “operationalized” process?

- The people who use the process own the process and have the authority to adapt and improve it.
- The “process owners” are in the best position to understand the process strengths and weaknesses.
- If people “own the process,” they will be more willing to fairly evaluate process changes.

Once you collect data, what do you do with it?

Discussion:

- Why do you need to periodically review your process data?
- How often should you review your process data?
- What happens if you review your process data too often? too seldom?

If you have already set goals, you start by understanding your performance against those goals.

Analyzing Performance

Analyze your performance with respect to size

estimation, effort estimation, and quality management to:

- understand your current performance
- identify your highest-priority areas for improvement
- establish challenging but achievable goals, and
- define corresponding improvement actions to meet those goals
- define actions to address challenges you will face in making those changes

Analysis of Size Estimating Accuracy

Review your performance on size estimating accuracy. For example:

- How much did your size estimating accuracy change? Why?
- Do I have a tendency to add/miss entire parts?
- Do I have a tendency to misjudge the relative size of parts?
- Do I need to calculate relative size range data using my historical data?
- Based on my historical size-estimating accuracy data, what is a realistic size-estimating goal for me?
- How can I change my process to meet that goal?

Analysis of Time Estimating Accuracy

Review your performance on effort estimating accuracy. For example:

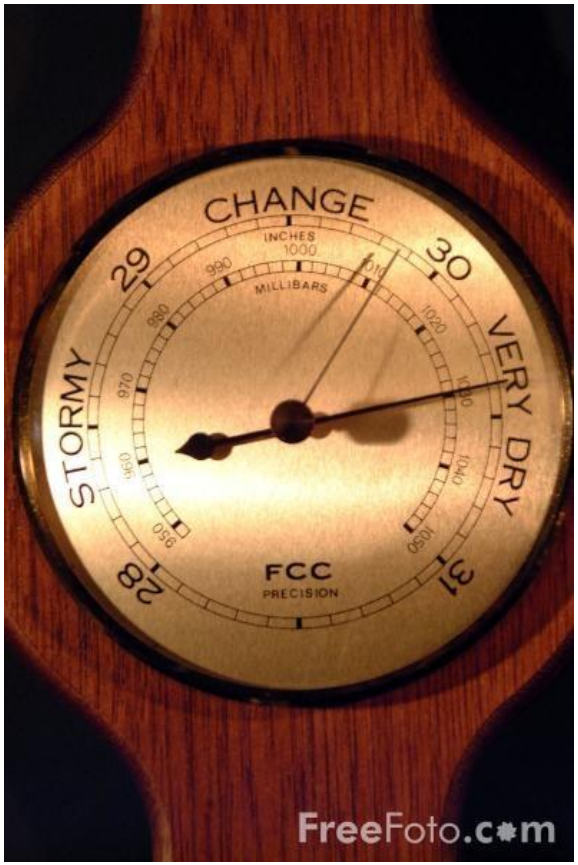
- How much did your effort estimating accuracy change? Why?
- Is my productivity stable? Why or why not?
- How can I stabilize my productivity?
- How much are my time estimates affected by the accuracy of my size estimates? (Would multiple regression help me?)
- Based on my historical time-estimating accuracy data, what is a realistic time-estimating goal for me?
- How can I change my process to meet that goal?

Defect and Yield Analysis

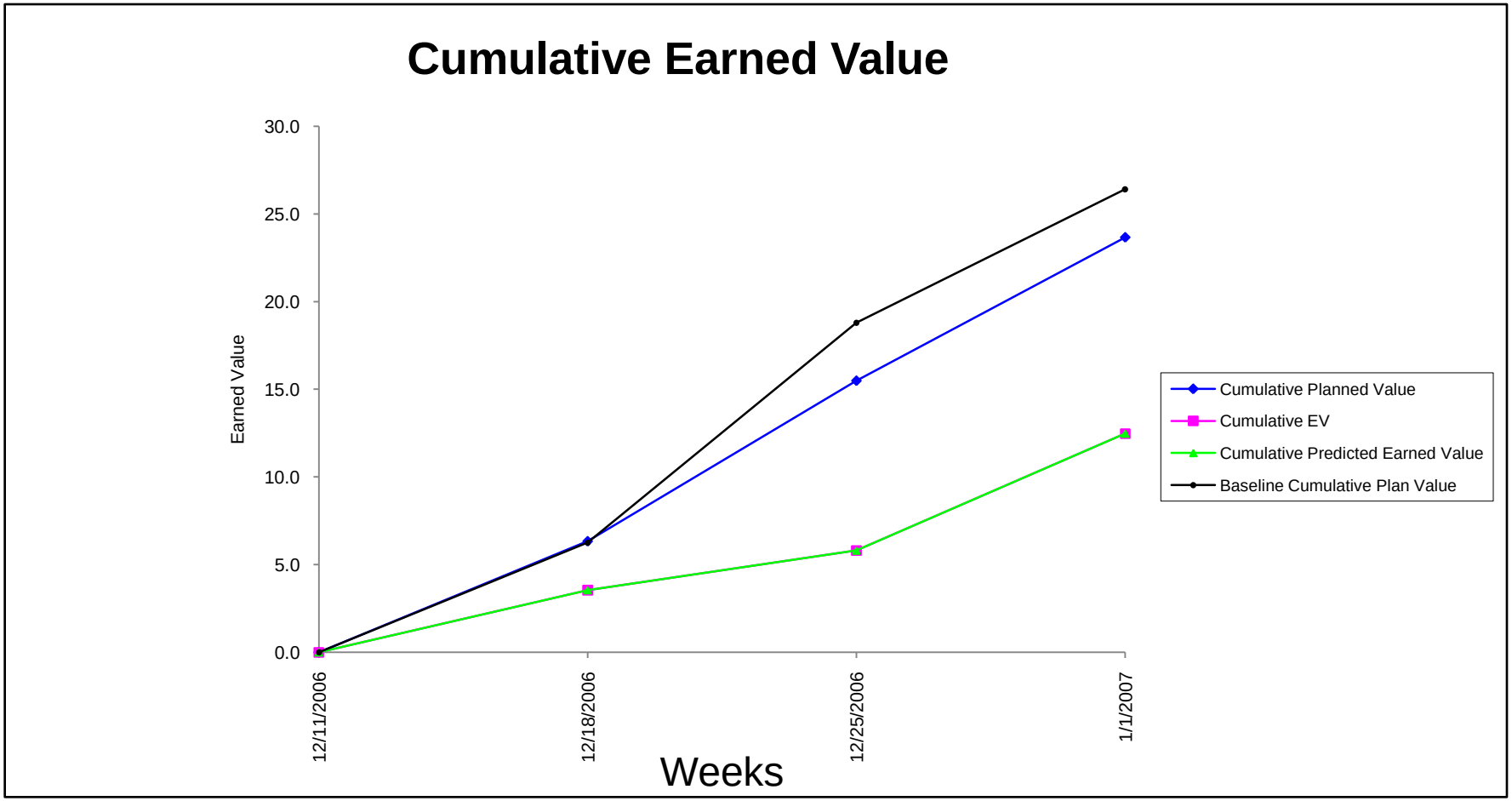
For example:

- What type of defects do I inject during design and coding?
- What trends are apparent in defects per size unit (e.g., KLOC) found in reviews, compile, and test?
- What trends are apparent in total defects per size unit?
- How do my defect removal rates (defects removed/hour) compare for design review, code review, compile, and test?
- What are my review rates for design review and code review?
- What are my defect-removal leverages for design review, code review, and compile versus unit test?
- Is there any relationship between yield and review rate for design and code reviews?

Leading vs. Lagging Indicators



Case Study - 1



Case Study - 2

TSP Week Summary - Form WEEK

Name	Consolidation						**Date**	11/8/2007	
Team	Example week								
Status for Week	2	**Selected Assembly**					**Cycle**	1	
Week Date	7/2/2007	SYSTEM							
						Plan / Actual	**Plan - Actual**		
Task Hours %Change		**Weekly Data**	**Plan**	**Actual**				**Project End Dates**	
Baseline	1280.1	Schedule hours for this week	45.5	26.9		1.69	18.6	Baseline	2/4/2008
Current	1332.1	Schedule hours this cycle to date	86.9	48.6		1.79	38.3	Plan	2/4/2008
%Change	4.1%	Earned value for this week	1.3	0.7		1.86	0.6	Predicted	11/16/2009
		Earned value this cycle to date	3.7	3.4		1.10	0.3		
		To-date hours for tasks completed	44.7	31.9		1.40			
		To-date average hours per week	43.4	24.3		1.79			
		EV per completed task hour to date	0.075	0.105					

A team is in week 2 of 7 month plan.

The team is behind 10% in Earned Value but the projected date for project completion is 2 years late— what is the problem?

The team on average is only getting a little more than half of their planned on-project task hours.

(1) Understand why the predicted project completion is two years late?

(2) Why aren't team members achieving planned on-project task hours?

Case Study – How Do You Get This Information?

From having operationally, defined processes (e.g., development process)

From basic, measurement data

- Operational measures (size, effort, schedule, quality)
- Measurement Definitions (task hour, defect, …)

From tools

- To record and analyze data

From having a realistic plan

- Developed by team members who use their own data for estimating and planning

Operational Definition

Task Hour

- Count effort applied to a specific project task
- Do not count
  - Break time
  - Project tasks not in the earned value plan
  - Non-project tasks

Operational Definition

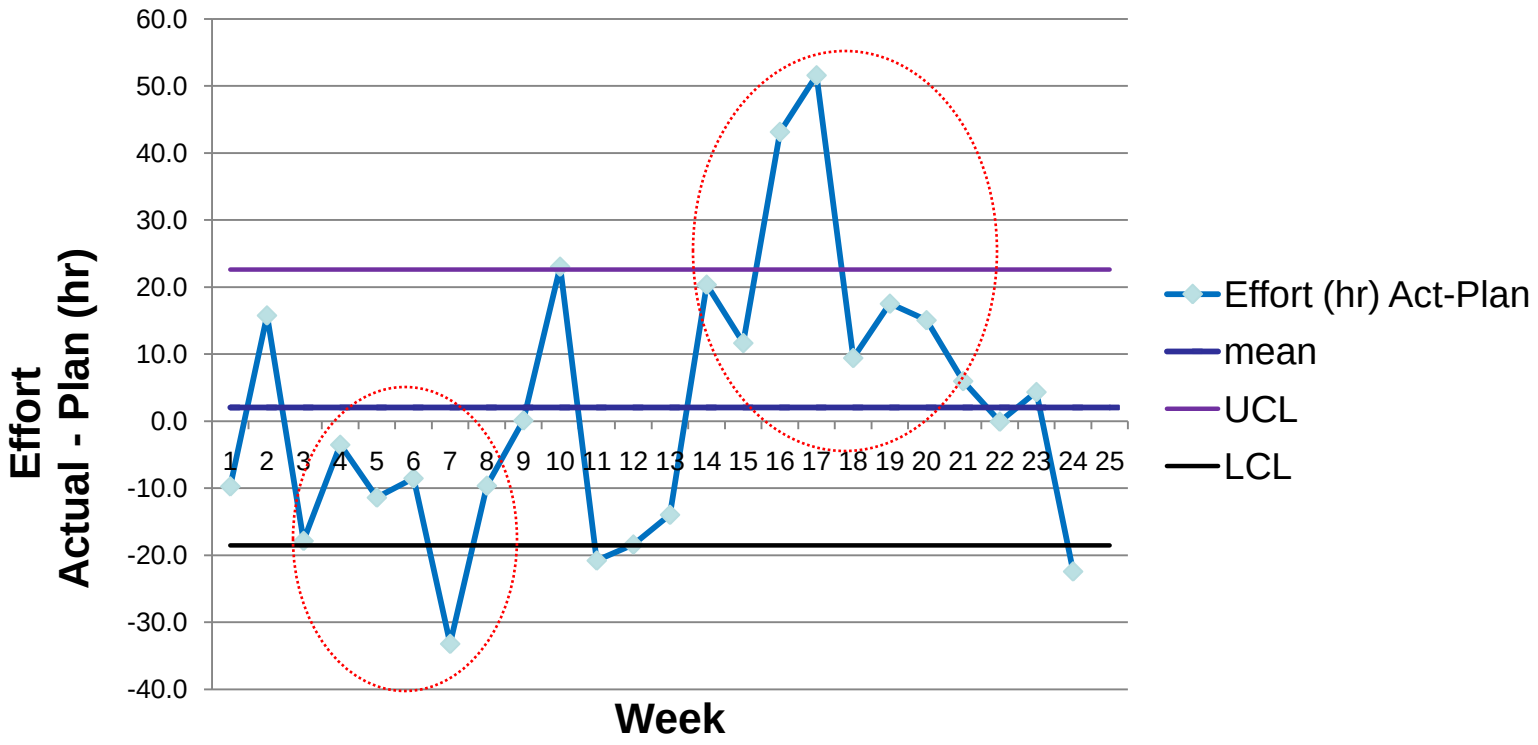
Earned Value

- Planned Value for task = estimated effort (cost) for task divided by sum of estimated effort for all project tasks

- Earned Value credited when task is complete

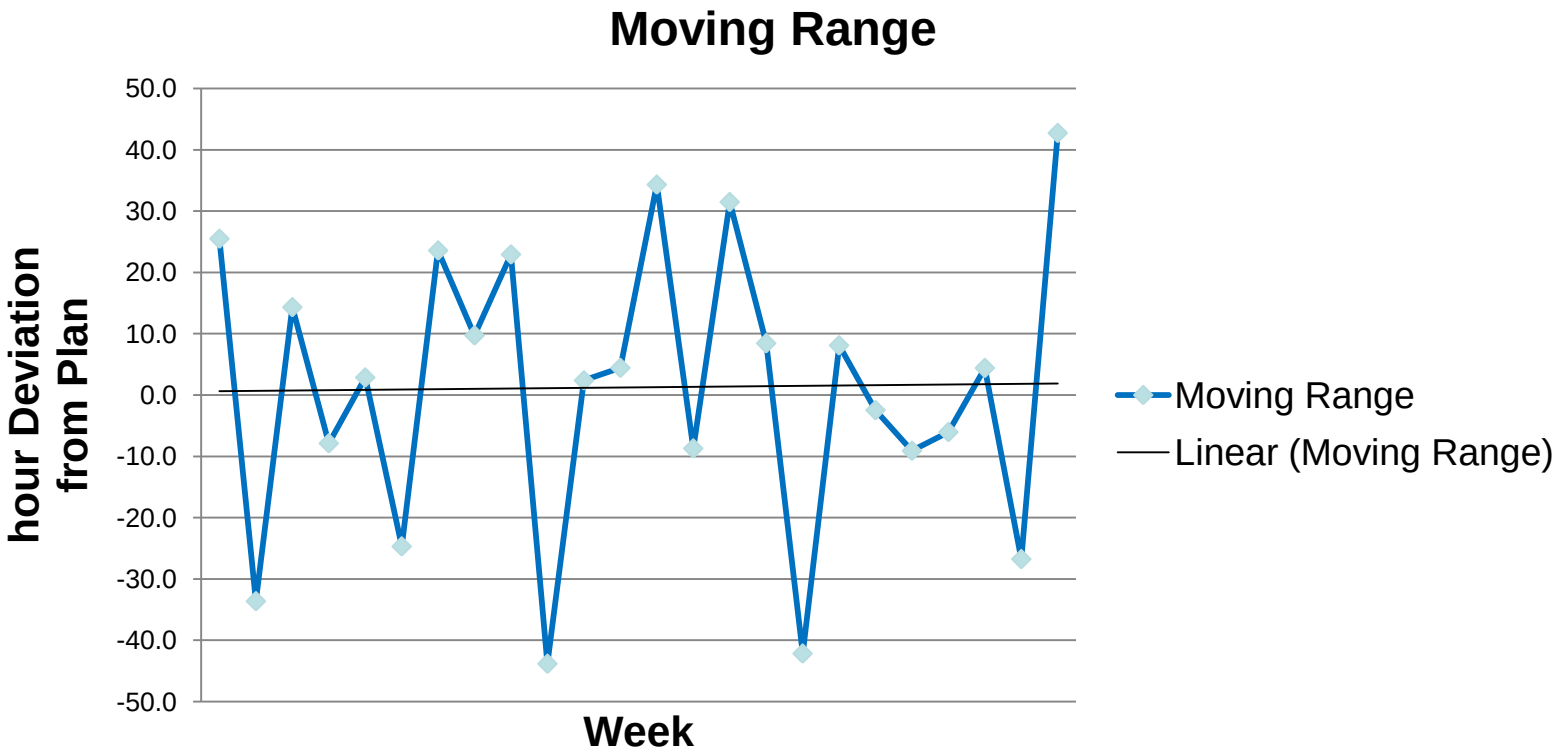
- In this definition Earned value always approaches 1.0 as the project nears completion

Each Week: (Actual – Planned) Effort [hours]

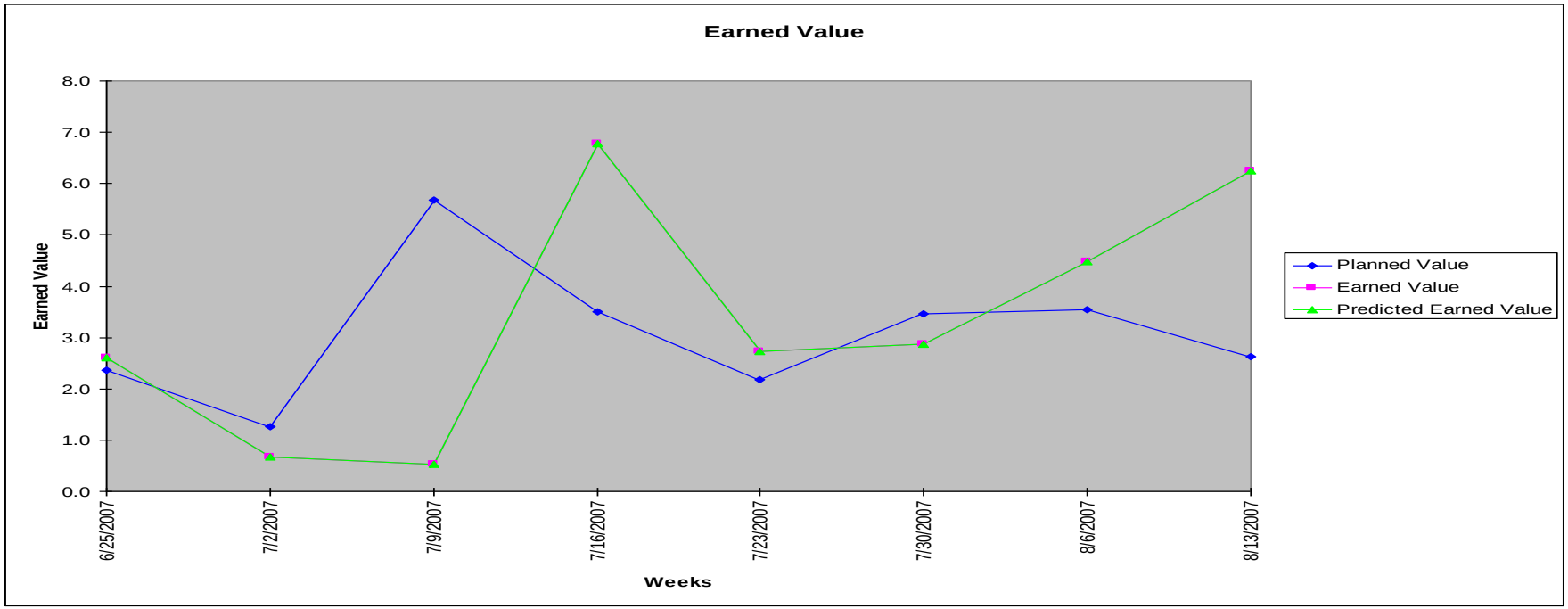


The team addressed the project effort problem.

Variation



Week 8, Schedule Progress (Earned Value)



After initially falling farther behind, weekly progress stabilizes.

Weekly Status Report

Weekly status reviews:

- Plan assumptions
  - Effort plan
  - Upcoming work tasks
- Project status
  - Actual effort
  - Earned Value
  - Cost Performance
- Projections based on status and history

Week 8 Team Report

TSP Week Summary - Form WEEK

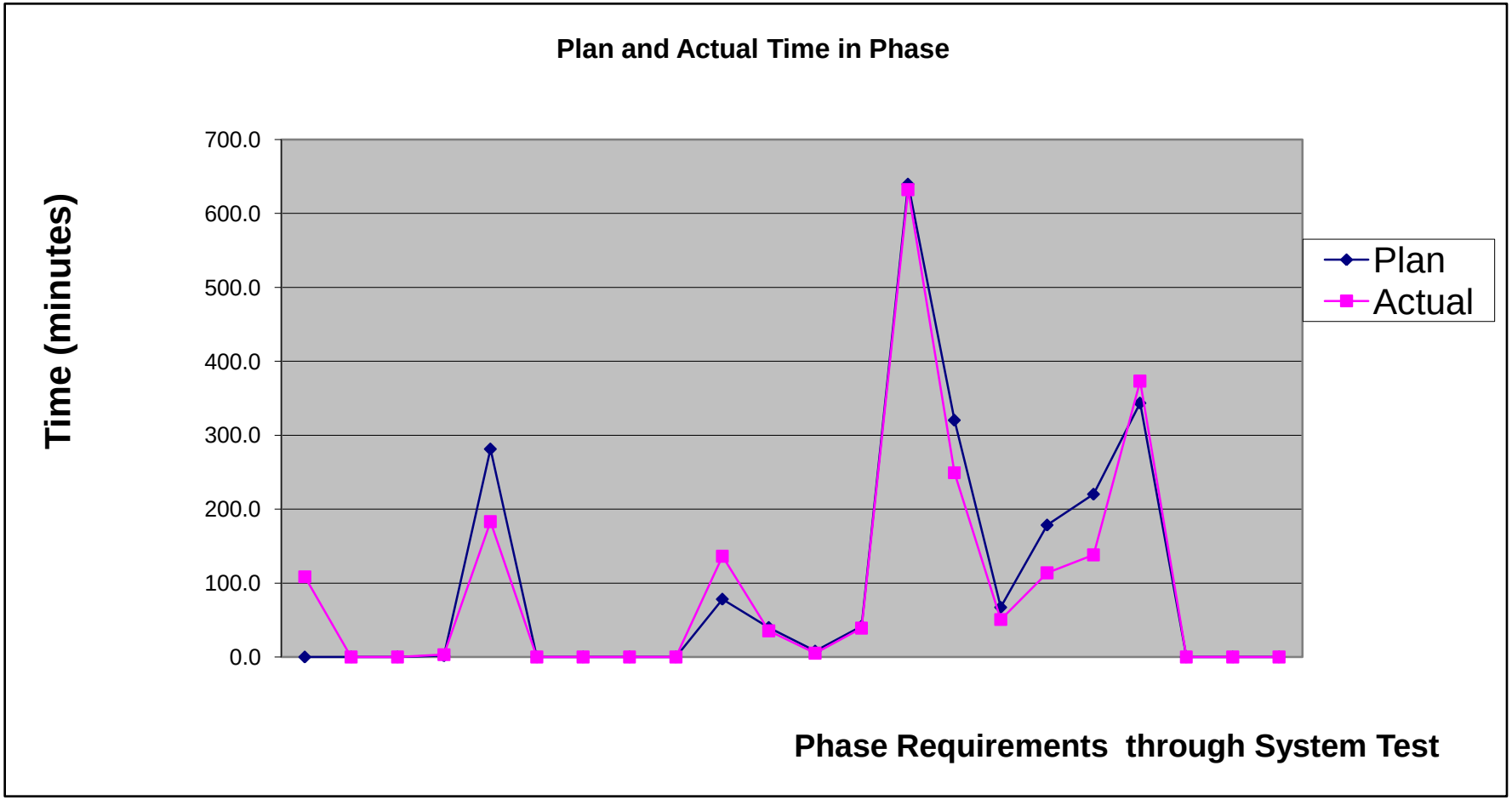
Name	Consolidation					**Date**	11/8/2007	
Team	Example week 8							
Status for Week	8	**Selected Assembly**				**Cycle**	1	
Week Date	8/13/2007	SYSTEM						

Task Hours %Change		**Weekly Data**	**Plan**	**Actual**	**Plan / Actual**	**Plan - Actual**	**Project End Dates**	
Baseline	1280.1	Schedule hours for this week	47.3	43.8	1.08	3.4	Baseline	2/4/2008
Current	1358.8	Schedule hours this cycle to date	364.1	306.7	1.19	57.3	Plan	2/4/2008
%Change	6.1%	Earned value for this week	2.6	6.3	0.42	-3.6	Predicted	4/21/2008
		Earned value this cycle to date	24.6	26.9	0.91	-2.3		
		To-date hours for tasks completed	365.7	293.1	1.25			
		To-date average hours per week	45.5	38.3	1.19			
		EV per completed task hour to date	0.074	0.092				

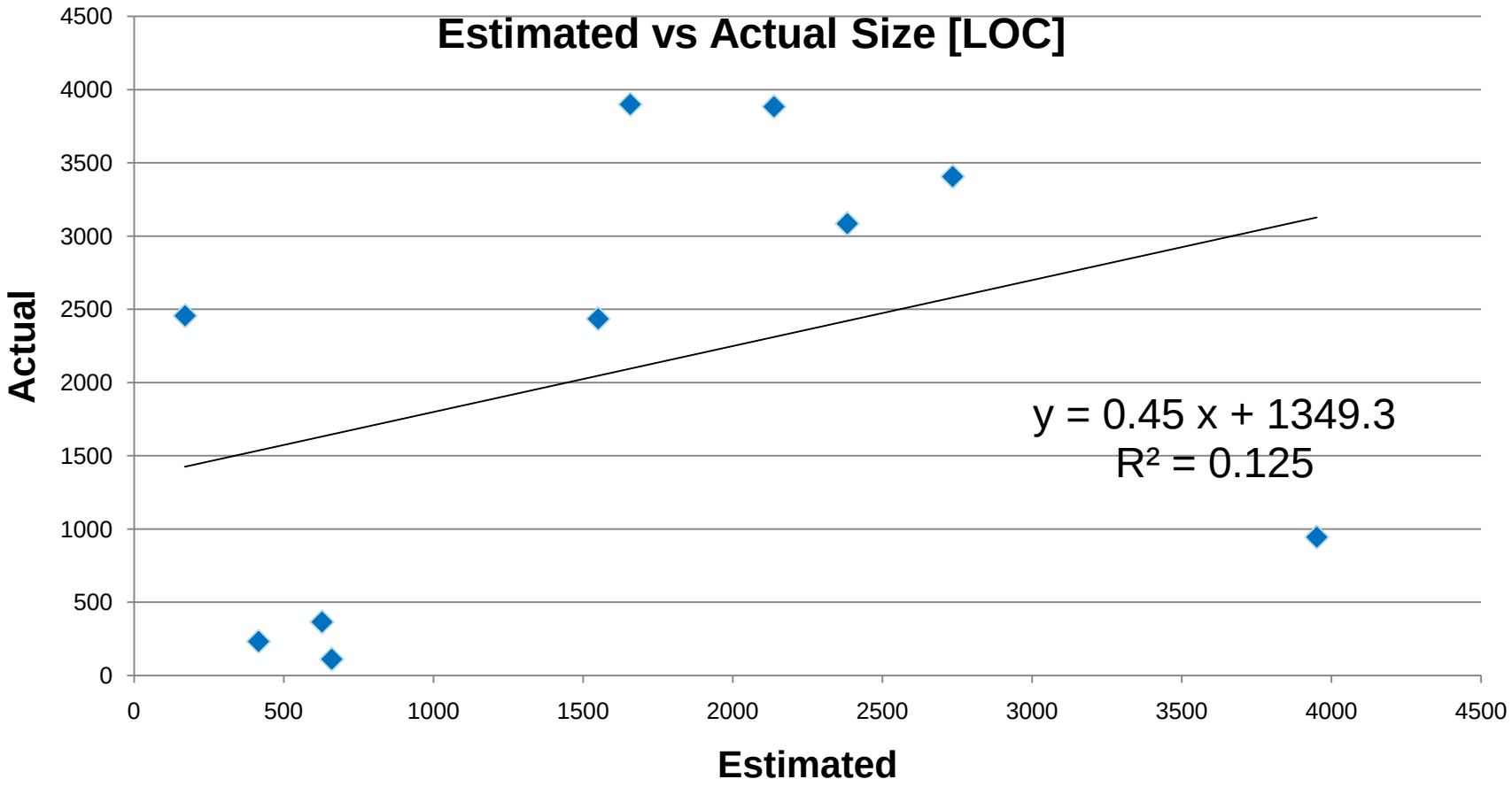
The team actions have been effective:

- Cumulative hours have not caught up
- The team is 9% ahead of schedule
- The predicted end date is now 2 months late rather than 2 years

Are They Following Their Process?



Size Estimation



Case Study Wrap-up

Teams and individuals need to assess performance with respect to goals:

- Did we achieve our performance goals? Why or why not?
- Where do we need to improve? What could we do differently? How would it change our performance?
- What kind of analyses need to be performed?

Summary

Build high performance through teams

Enable high maturity capabilities by building a solid foundation

CMMI and TSP are mutually reinforcing—

- CMMI provides the principles for process improvement and organizational focus
- TSP can be useful for providing team discipline and operationalizing CMMI practices

Questions?

Tim Chick

tchick@sei.cmu.edu

412-268-1473

PSP/TSP website:
http://www.sei.cmu.edu/tsp

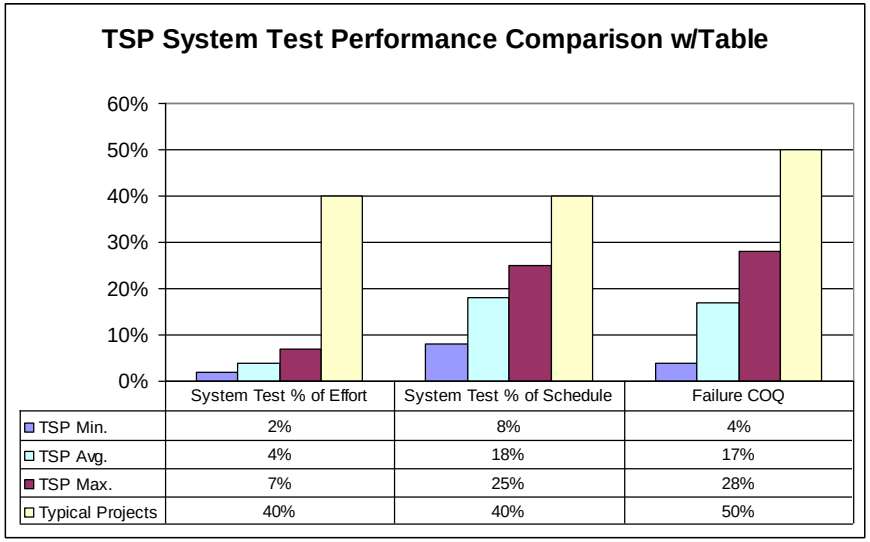
NAVAIR Benefits from TSP

Program	Size of Program	Defect Density (Defects/KSLOC))	Cost Savings from Reduced Defects
AV JMPS	443 KSLOC	0.59	$2,177,169
P-3C	383 KSLOC	0.6	$1,478,243

Program	Schedule Variance	Cost Variance
AVJMPS	0.5% overrun	1.5% overrun
H2.0	1.1% overrun	6.9% overrun

Quality Benefits

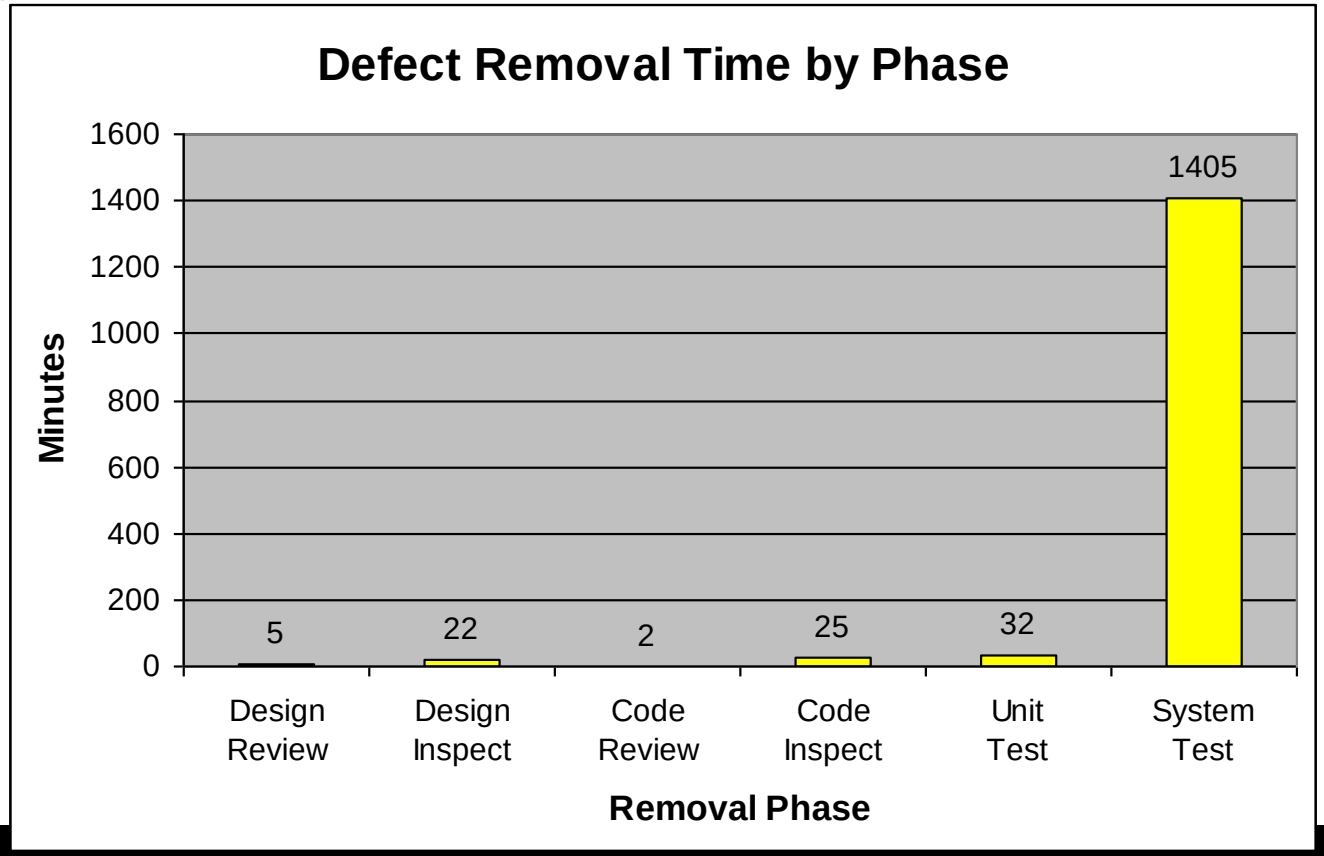
- TSP dramatically reduces the effort and schedule for system test.
- Most defects are removed during reviews and inspections at a cost of 2 to 25 minutes per defect.
- System test removal costs run from to 2 to 20 hours per defect.
- These benefits continue after delivery.
  - lower support costs
  - satisfied customer
  - better resource utilization



	System Test % of Effort	System Test % of Schedule	Failure COQ
TSP Min.	2%	8%	4%
TSP Avg.	4%	18%	17%
TSP Max.	7%	25%	28%
Typical Projects	40%	40%	50%

Reviews and Inspections Save Time

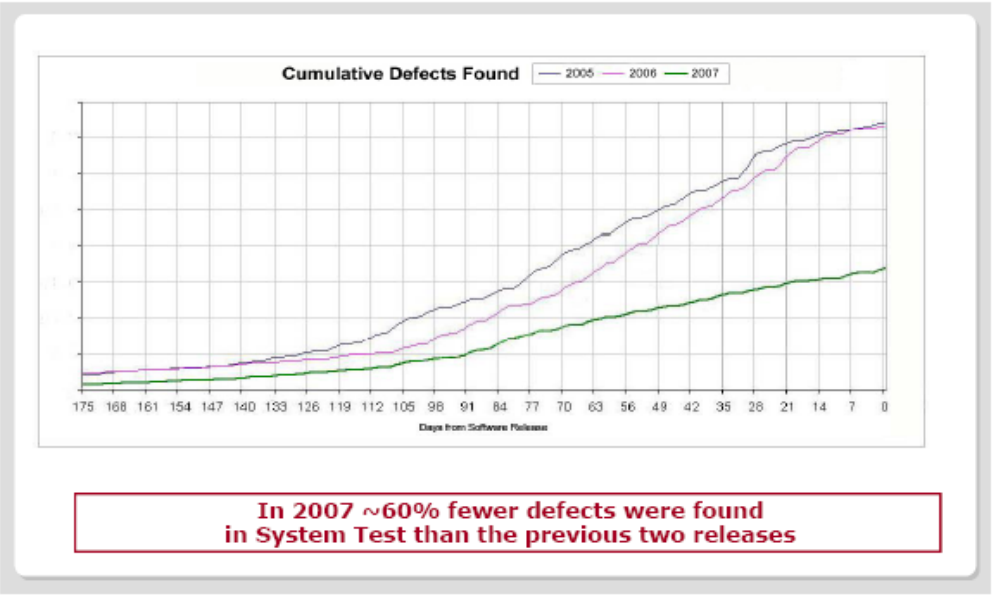
- Xerox found that TSP quality management practices reduced the cost of poor quality by finding and removing defects earlier when costs are lower.



Intuit Quality Improvement

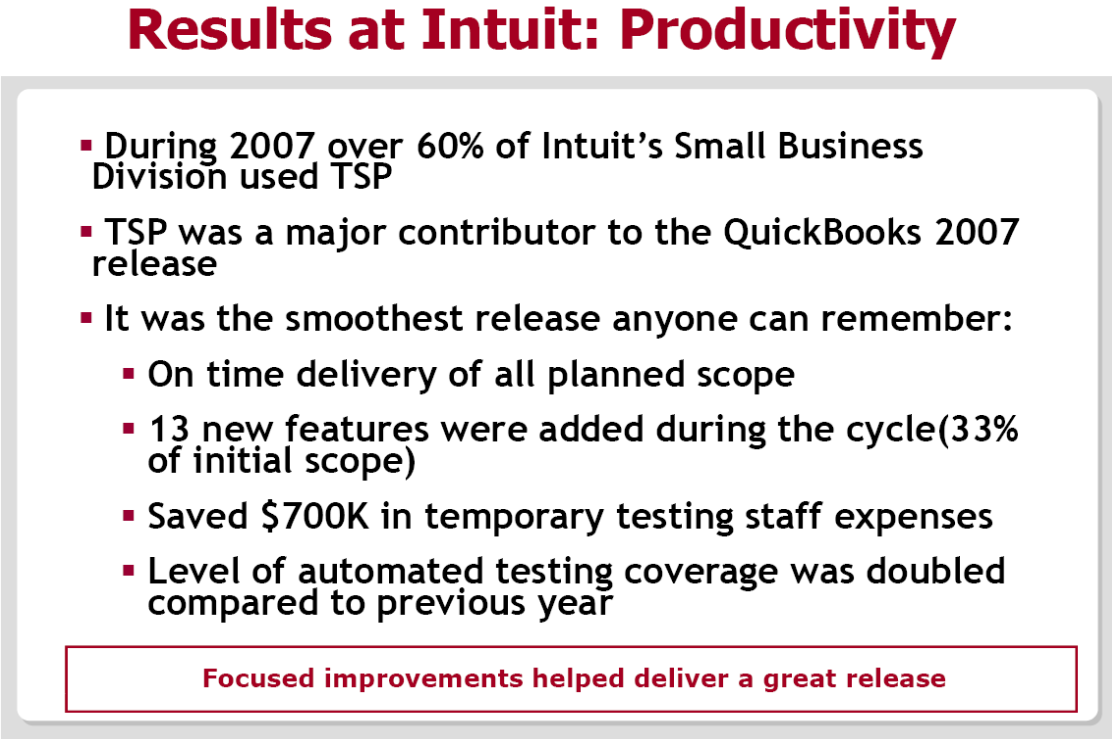
- TSP reduced defects found in system test by 60% over the previous two releases of QuickBooks 2007 release.
- Intuit has also recently reported a savings of $20M from a reduction in customer support calls on QuickBooks 2007.

Results at Intuit: Improved Quality



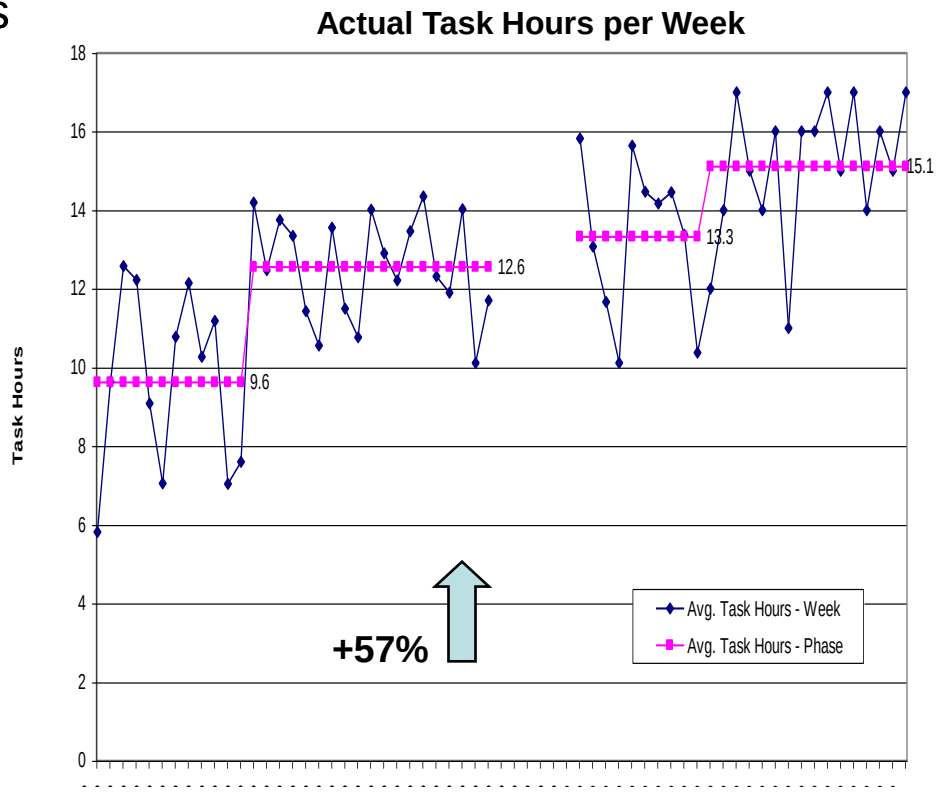
Intuit Productivity Improvement

- By putting a quality product into system test Intuit improved productivity and reduced cost while delivering 33% more functionality than planned.



Improving Task Hours

- At Allied Signal average task hours per developer per week were improved from 9.6 hours to 15.1 hours through quiet time, process documentation, more efficient meetings, etc.
- This is equivalent to a 57% increase in productivity.
- If you didn't have such detailed information, would you even know that you had a problem? Or an opportunity for such dramatic improvement?



Source: Allied Signal

Intuit Test Schedule Reduction

- From data on over 40 TSP teams, Intuit has found that
  - post code-complete effort is 8% instead of 33% of the project
  - for TSP projects, standard test times are cut from 4 months to 1 week
- Testing time is reduced from four months to one month.

Microsoft Schedule Improvement

- First-time TSP projects at Microsoft had a 10 times better mean schedule error than non-TSP projects at Microsoft as reflected in the following table.

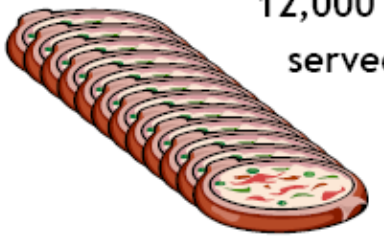
Microsoft Schedule Results	Non-TSP Projects	TSP Projects
Released on Time	42%	66%
Average Days Late	25	6
Mean Schedule Error	10%	1%
Sample Size	80	15

Work-Life Balance

- People are your most important resource.
- Finding and retaining good people is critical to long-term success.
- Intuit found that TSP improved work-life balance, a key factor in job satisfaction.

Results at Intuit: Improved Work-Life Balance

- Half as many weekend source check-ins (<3%)
- Reduced $ on dinners as measured by PSS - "Pizza Slices Served"



12,000 pizza slices served last year

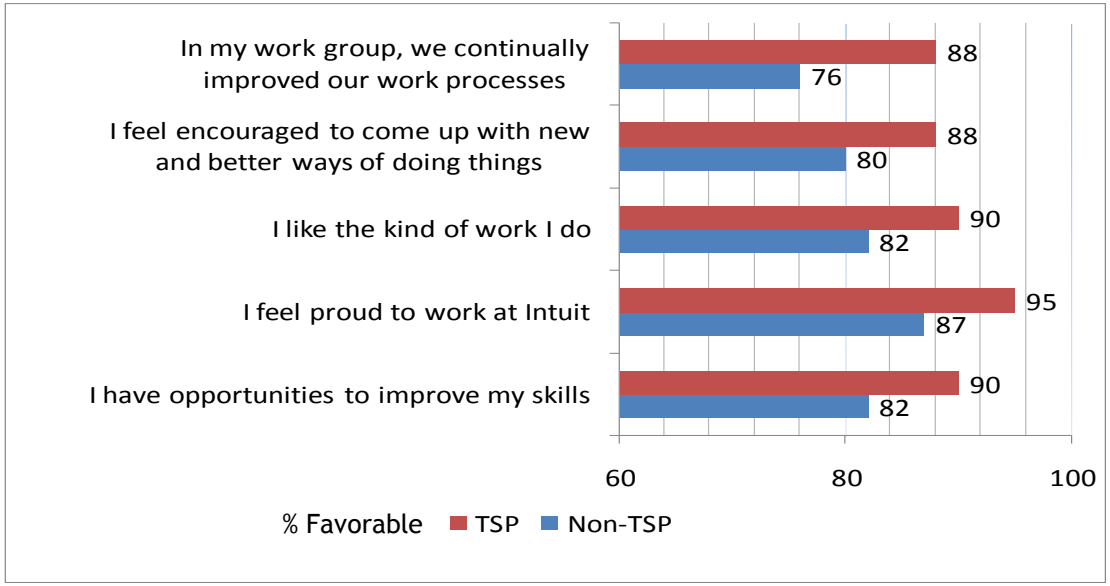
vs

~30 pizza slices this year

TSP helped improved employee work life balance

Intuit TSP Survey Results

- Improved work-life balance with TSP is reflected in job satisfaction surveys.



"Engineers love it… Once they adopt it they can't imagine going back"

Source: Intuit