CMMI-Based Process Improvement: How and When Does Success Happen?

Dennis R. Goldenson
Software Engineering Institute

Angel Liu (Liu Qi) & Qi Jianping
Motorola Software Group, China Center

CMMI Technology Conference
14 November 2006

Acknowledgments

Special thanks are due to Sasha Babkin, Ruth Buys, Craig Hollenbach, Larry McCarthy, Pete McLoone, Lynn Penn, Rolf Reitzig, Debra Roy, Millee Sapp, Bob Stoddard & Bob Weiser. This work would not be possible without their willingness to share both their data & ideas.

The presentation is greatly enhanced by the help graciously provided under difficult circumstances by my co-authors Angel Liu (Liu Qi) & Qi Jianping from the Motorola Software Group China Center.

Today's Talk

Setting the stage

Our approach

Current & Proposed Studies

Enterprise wide performance benchmarking

Example results

How can you help?

Setting the Stage

Purpose: Describe Ongoing Research Meant To...

1. Provide objective evidence to substantiate value added by CMMI-based process capabilities
   - To what extent are there better performance outcomes when organizations & their projects follow processes that satisfy the goals of CMMI?
2. Improve processes & resultant performance outcomes for organizations that participate in this research
   - As well as for others that modify their own measurement activities similarly to support performance driven improvement
3. Explain less than stellar product delivery by high maturity organizations as well as low
   - As well as process enactment at the program/project level that does not always match appraised maturity
   - Document the extent of such occurrences

Why is this Work Important?

Substantial proof of concept exists

- CMMI-based process improvement can & has led to concomitant improvement in performance outcomes
- Predictably faster, better, cheaper product development and maintenance

But skepticism remains about the value of disciplined adherence to well defined processes

- As opposed to solutions *de jour*
- That are ***not*** necessarily at odds with processes that satisfy the goals of CMMI best practices – e.g., Agile or Six Sigma methods

Some Existing Evidence

Performance Results of CMMI-Based Process Improvement, Technical Report, CMU/SEI-2006-TR-004, August 2006

Benchmarking CMMI Cost and Impact: Interim Report, December 2004 (Distribution of full document limited to benchmark contributors)

Demonstrating the Impact and Benefits of CMMI: An Update and Preliminary Results, Special Report, CMU/SEI-2003-SR-009, October 2003

CMMI Performance Results Web site

- http://www.sei.cmu.edu/cmmi/results.html

Numerous presentations

- At this & previous conferences

What's Missing?

Most existing work is based on case studies

- Which are based on credible quantitative evidence
- But circumstances differ
  - Case based results ***can*** be very instructive
  - But, they may ***not*** be applicable elsewhere
  - & they can be accused of "cherry picking" ... fairly or not

More generalizable comparative analyses are needed ... ***Process models!***

- Of the effects on performance outcomes of ***differences*** in process enactment
- Under ***varying*** organizational circumstances & product characteristics
  - ***That may affect variation in both successful process enactment & performance outcomes***

What Else is Needed?

A viable benchmarking infrastructure & community of practice

- In a field where people aren't comfortable sharing information
- Without which we're just guessing about "industry standards" ...

Empirical analyses that focus on barriers & facilitators of adoption & improvement initiatives

- Process capability doesn't always guarantee successful program performance & product quality
- Not all improvement initiatives are implemented successfully

Today's Talk

Setting the stage

Our approach

Current & Proposed Studies

Enterprise wide performance benchmarking

Example results

How can you help?

Our Approach

Current State

Case studies

- With quantitative evidence of process improvement & concomitant performance gains
- Typically showing total results over time ... often rolled up over multiple projects

Often accompanied by qualitative affirmations

- Based on experience of those doing the work
- That process improvement is the major source of change

But little explicit discussion of competing explanations

- Whether or how they were considered
- Leading to accusations of spurious correlation

Understanding Variation & Causal Thinking

There almost ***always*** is variation

- In product quality, project & organizational performance
- How processes are enacted
  - The existence of defined processes
  - Adherence/compliance with them
  - & how well the processes are enacted ... the "goodness" issue
- & the other factors that may effect both process & performance outcomes

Looking only at total results

- May mask important differences at the project level ... where most of the development work actually takes place
- {Per an important statistical literature about possible misinterpretations from looking only at the marginal totals as opposed to individual data points}

Bases for Comparison

Organizations vary widely in the measures that they use

- & their use of measurement altogether

No industry standard measurement criteria yet exist

- Even if they did exist ... organizations in our field are cautious to say the least about sharing proprietary information about their business assets

Organizations differ widely in ***how*** they establish processes that satisfy the goals of CMMI

Other factors must be considered in addition to process & performance

- Staff capabilities (skills & expertise), tools & technologies used, product domains, precedentedness, interactions with acquirers & so on
- All of which can & should be the subject of defined processes

Our Evolving Research Agenda

A two-fold approach

1. Examine performance effects of variation in maturity level & capability level
   - For that we need wider variation in process enactment
   - Results most pertinent for organization less far along in their improvement journeys
2. Higher maturity organizations (by definition) have (or should have) less variation in ***what*** processes they follow
   - But there can be, & sometimes is, variation in ***how*** they do it
   - & some variation ***does*** remain in their performance outcomes

Today's Talk

Setting the stage

Our approach

Current & Proposed Studies

Enterprise wide performance benchmarking

Example results

How can you help?

Current & Proposed Studies

Systems Engineering Effectiveness

Survey of DoD Program contractors

- In collaboration with NDIA Systems Engineering Division
- Under strict conditions of non disclosure

Measures of:

- Earned value, other performance criteria, project & product context
- Process capability questions about existence, use & quality of interim process work products
- Consistency with appraised maturity levels

Results currently expected in 1st quarter of CY2007

Performance Measurement Associated with DoD Program Monitoring

Independent judgments about programs' adherence to processes consistent with appraised maturity of parent organizations

- Program monitoring by DCMA ... along with surveys of performance outcomes
- Currently considering similar analyses with ~20-30 programs for which monitoring appraisals have been done by SEI ... & possibly others of comparable quality

Possibly in concert with a "SCAMPI+" performance module

- Appraising project/program performance
- Linked directly with process appraisal results

The Performance Benchmarking Consortium (PBC)

Members with existing quantitative measures & experience using them

- Working together to create (& then enhance) a common product suite

Measures & a repository infrastructure to

- Facilitate benchmarking comparisons
- Provide CMMI-based measurement guidance
- Supported by appropriate trailing activities

Come to Mark Kasunic’s presentation for much more detail

Today’s Talk

Setting the stage

Our approach

Current & Proposed Studies

Enterprise wide performance benchmarking

Example results

How can you help?

Enterprise Wide Performance Benchmarking

Overview$_1$

Work with larger enterprises

- Comprised of multiple projects, contractual programs, & other constituent organizational units
- That already have &/or are evolving common measures of performance
- Along with complimentary measures of process enactment, organizational & product attributes

An important way to provide:

- More generalizable, comparative analyses explaining variation
- In both successful process enactment & the performance outcomes that the processes are meant to achieve

Especially important *in lieu* of shared measures that enable wider state of the practice & benchmarking analyses

Overview$_2$

Participating organizations bring:

- Validated data that are actually used to inform management & technical decisions
- Personnel and financial resources to support the analyses

Work now in progress with three selected organizations

- All want to focus on factors that affect that performance
- Also prefer to address their own more concrete “how to” questions
- Augmented with survey or other mining of “tribal knowledge” about current variations in process enactment that may account for exemplary outliers
- Initial focus on high maturity organizations within their larger enterprises
- All three do have somewhat wider capability variation though
  - & BAE has wider maturity level variation

Overview$_3$

Current Participants

- Motorola Software Group
- Lockheed Martin Integrated Systems & Solutions
- BAE

Work also provides useful how-to guidance for others elsewhere

- With enough detail to give others a good sense of the participating organizations' processes without divulging anything proprietary

Work in progress

- Only a few example results shown here
- More to follow next year's conference & in the interim

Similar Work$_1$

Similar studies exist ... Mostly based on the SW-CMMI

- Clark, B. *The Effects of Software Process Maturity on Software Development Effort*. PhD Thesis, University of Southern California, April 1997.
- Deephouse, C.; Goldenson, D.; Kellner, M.; & Mukhopadhyay, T. “The Effects of Software Processes on Meeting Targets and Quality,” 710-719. *Proceedings of the Hawaiian International Conference on Systems Sciences*. Wailea, HI, Jan. 3-6, 1995. Los Alamitos, CA: IEEE Computer Society Press, 1995.
- K. El Emam, A. Birk (2000). ‘Validating the ISO/IEC measure of software development process capability’. *Journal of Systems and Software* 4(2), 113-133.
- El Emam, K. & Goldenson, D. “An Empirical Review of Software Process Assessments.” *Advances in Computers* 53 (2000): 319-423.
- Goldenson, D. & Herbsleb, J. *After the Appraisal: A Systematic Survey of Process Improvement, Its Benefits, and Factors that Influence Success* (CMU/SEI-95-TR-009, ADA 302225).

Similar Work$_2$

- Goldenson, D.; El Emam, K.; Herbsleb, J.; & Deephouse, C. “Empirical Studies of Software Process Assessment Methods.” in K. El Emam and N. H. Madhavji (eds.): *Elements of Software Process Assessment and Improvement*. Los Alamitos, CA: IEEE Computer Society Press, 1999.
- Harter, D. E.; Krishnan, M. S.; & Slaughter, S. A. “Effects of Process Maturity on Quality, Cycle Time, and Effort in Software Product Development.” *Management Science* 46, 4 (April 2000): 451-466.
- Jung, H. & Goldenson, D. *CMM-Based Process Improvement and Schedule Deviation in Software Maintenance* (CMU/SEI-2003-TN-015, ADA 416418).
- Krishnan, M.S. & Kellner, M. I. “Measuring Process Consistency: Implications for Reducing Software Defects.” *IEEE Transactions on Software Engineering* 25, 6 (November/December 1999): 800-815.
- Lawlis, P.; Flowe, R.; & Thordahl, J. “A Correlational Study of the CMM and Software Development Performance.” *CrossTalk* 8, 9 (September 1995): 21-25.
- Paulk, M.C. *An Empirical Study of Process Discipline and Software Quality*. Ph.D. Dissertation, University of Pittsburgh, 2005. <http://etd.library.pitt.edu/ETD/available/etd-070 82004-155917>

Higher Maturity Comparisons$_1$

Focus on finer grained how-to measures

- Either already existing or collected as needed ... perhaps with custom survey data
- As part of QPM, OPP, CAR & OID activities

CAR typically ...

- Looks for plausible explanations of special causes of variation
  - What's different about the exemplary & least successful projects
  - Both process & other factors
  - Where success is measured by performance outcomes
- Then defines or redefines processes, trains & implements them & monitors the performance effects of the changes in process

Higher Maturity Comparisons$_2$

Proactive, statistically based CAR

- Uses data from existing & enhanced measurement repositories
  - To examine as-is statistical relationships

Since there is less process variation in higher maturity organizations

- Probably makes more sense to look first for repeatable patterns between performance & ***other*** contextual factors
- Then identify, define, refine & evaluate the effects of new or changed processes

Where appropriate, use quasi experimental OID processes to:

- Pilot, incrementally deploy & modify the new processes as necessary based on measured performance results

Exactly what MSG China did

More “High Maturity” Measurement Guidance: OID

It is reasonable to pilot first with accommodating projects

- To refine new processes before wider deployment
- But need some degree of experimental control, e.g.,
  - Matching or paired comparisons
  - To attribute change to the process intervention as opposed to spurious correlation due to other factors
  - Then track & revise as necessary

Incremental deployment yields more confidence in cause & effect

- As opposed to “training pants” learning effects
- But need baselines of as-is state prior to process intervention

More "High Maturity" Measurement Guidance: CAR_1

"Cross sectional" statistical studies are less vulnerable to accusations of learning effects due simply to staff maturation over time

- But we do need process variation to separate process effects from competing explanations of differences in performance

Historical data in measurement repositories & other existing measurement infrastructure

- Make it easier for higher maturity organizations to make comparisons of otherwise comparable circumstances

The R in CAR still needs to be compared with an otherwise like baseline

- But that's possible when incremental changes are made with the same people, same product & project characteristics
- Especially after similar new projects apply the same new processes

More "High Maturity" Measurement Guidance: CAR_2

One can pool cases over time to increase variation in process enactment

- It's still necessary to control for staff experience & learning effects
  - Even though experience *per se* doesn't always explain all that much {as per the early research on programmer productivity}
  - Still, we need better measures of staff skill, e.g., design, code, domain knowledge
- With enough cases, the mix of contextual variables hopefully can be controlled statistically ... not just experimentally

Today’s Talk

Setting the stage

Our approach

Current & Proposed Studies

Enterprise wide performance benchmarking

Example results

How can you help?

Example Results

Cost of Quality at Motorola Software Group (MSG)

Cost of Quality (COQ) at MSG includes effort spent on...

- Review / Inspection
- Test development & execution
- Quality auditing, training, other process Improvement & problem prevention

Cost of Poor Quality (COPQ) includes...

- Rework & related failure correction throughout the life cycle

Both expressed as percentages of overall effort spent for product development

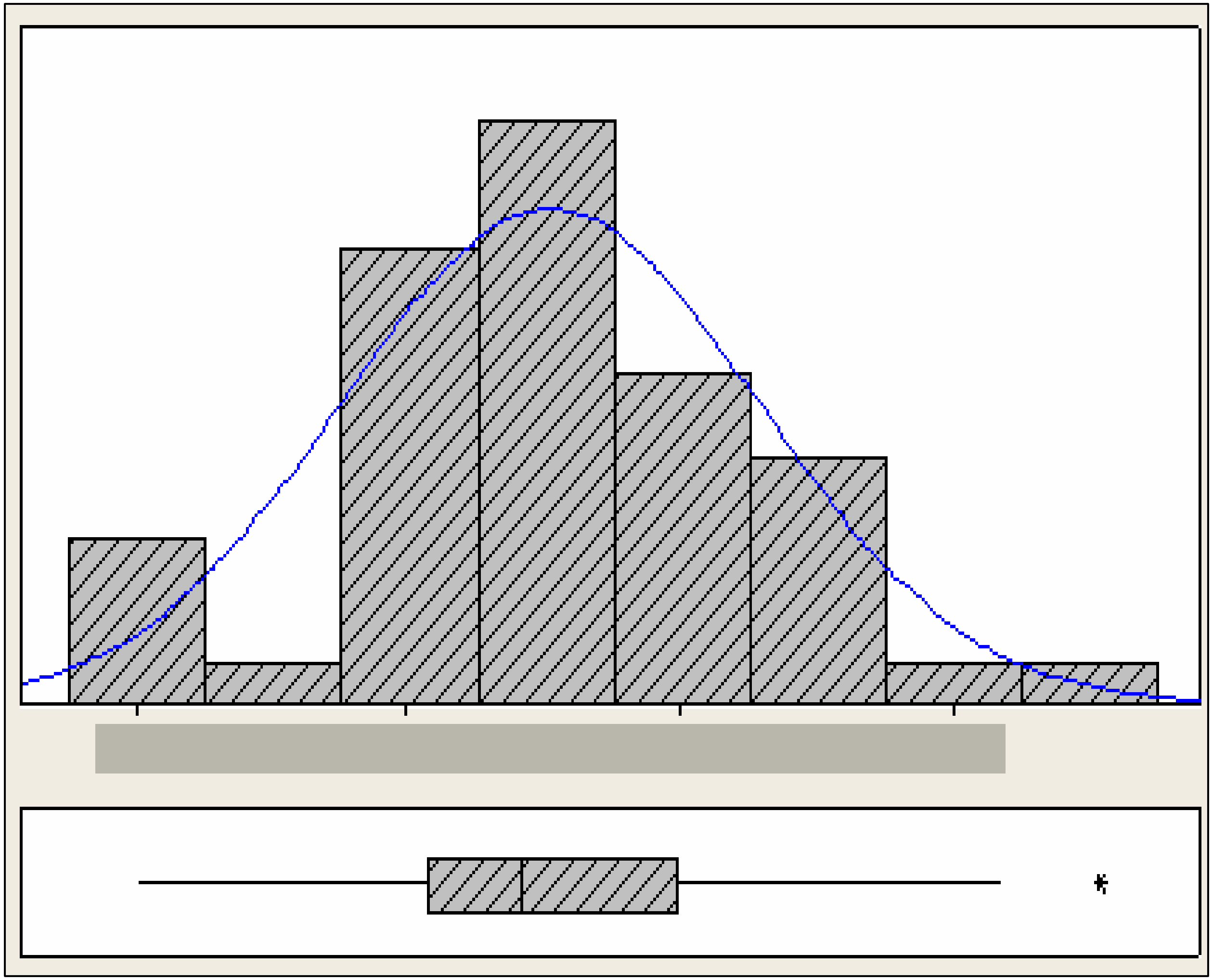
Results presented here examine ***test development & execution***

- Initial results from 2002 drill-downs ... N = 46 projects

Variation in Cost of Quality for Test Appraisal

Results from MSG China Center

- N = 46
- Actual values of data distribution (not shown here) are quite good by our sense of "industry standards"
- Still, there are variations that MSG wishes to reduce further



Differences by Domain

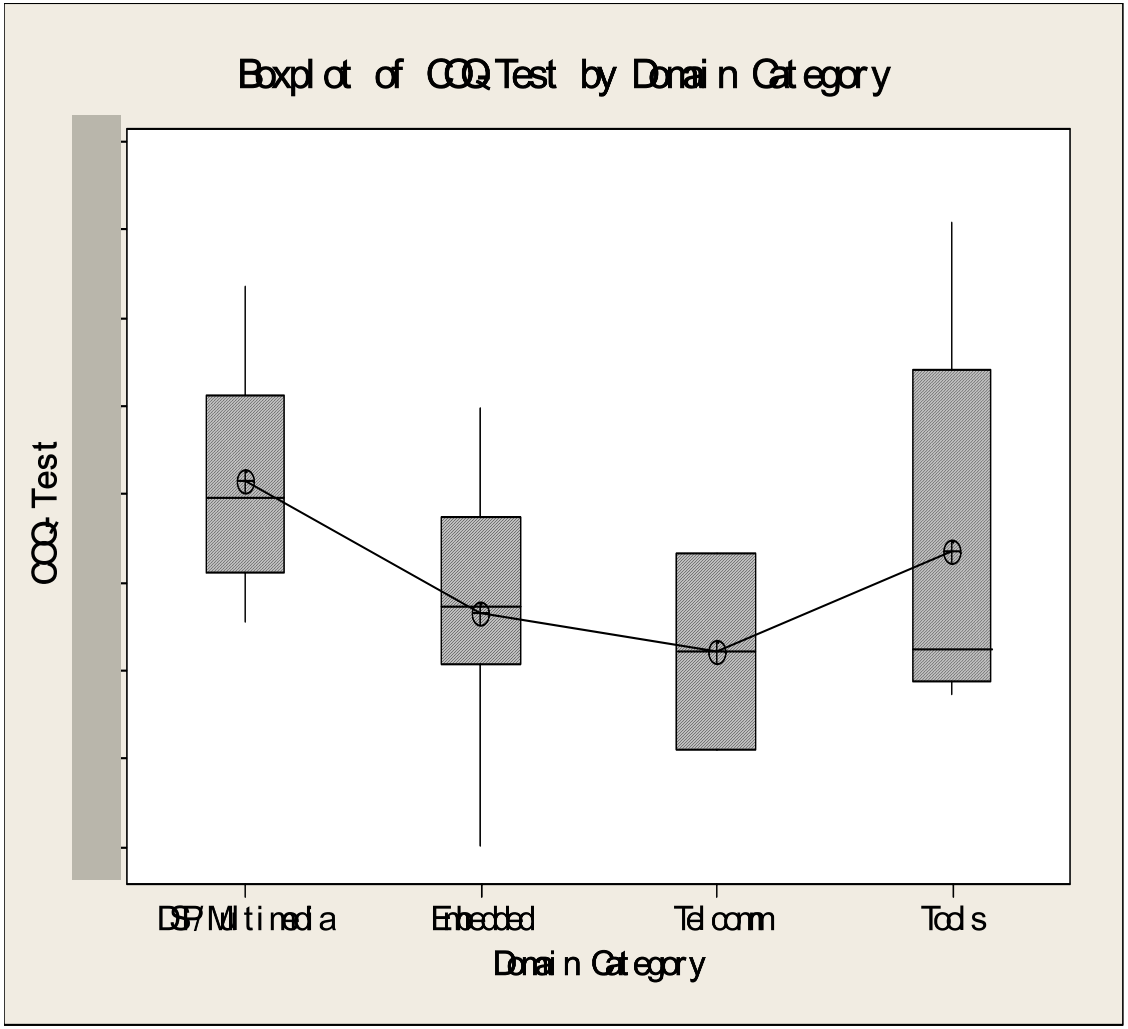
DSP/Multimedia projects have higher COQ-test

- Most Assembly projects fall here
- Some porting & optimization-oriented development projects, with high performance requirements
- & some product-oriented projects

Embedded vary more

Ns:

- 10 DSP Multimedia
- 29 Embedded
- 2 Telecom; 5 Tools

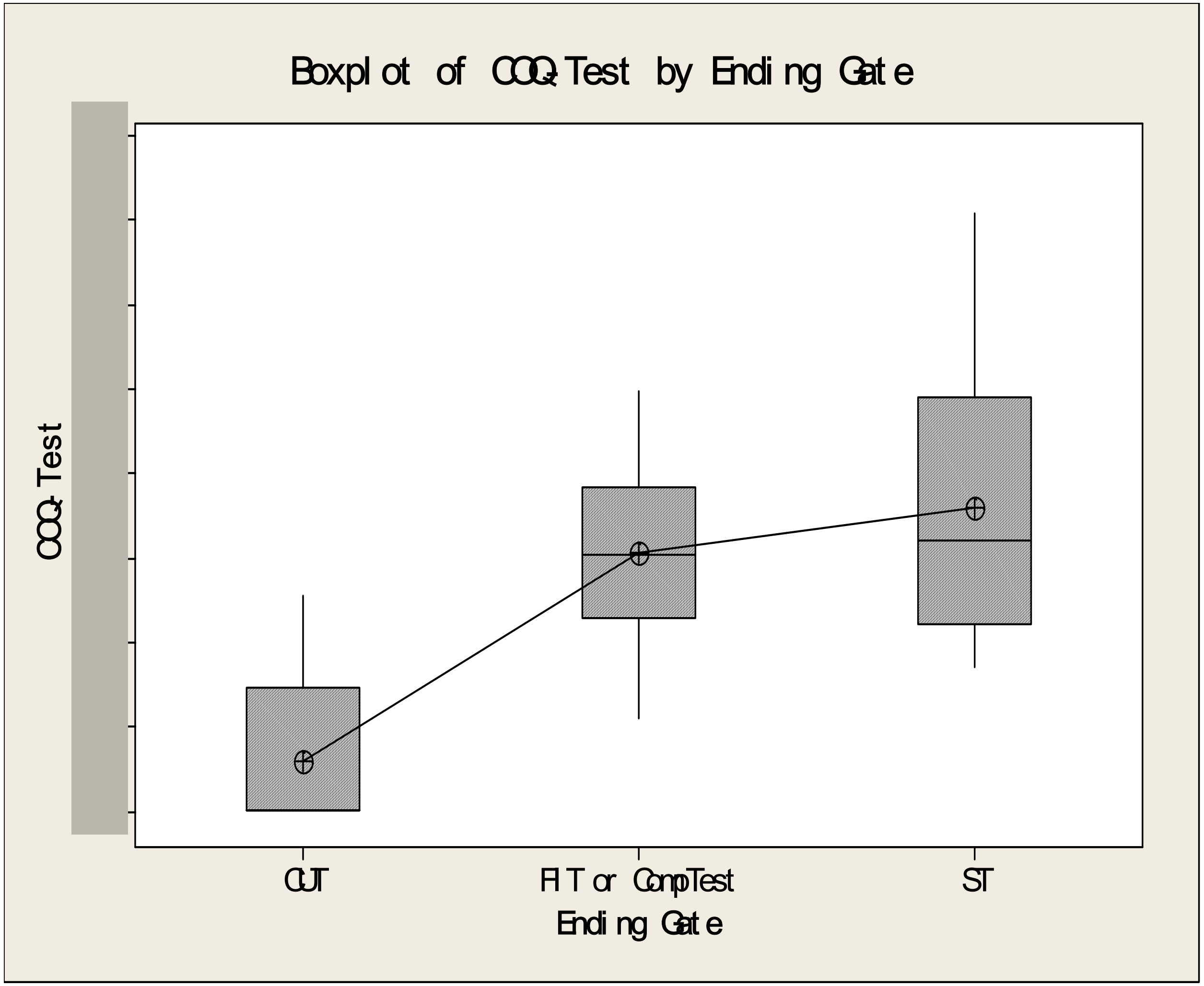


Differences by Motorola "End Gates"

Projects ending at System Test often have much higher COQ-Test

Ns:

- 5 Code & Unit Test (CUT)
- 20 Component Test & Feature Integration Test (FIT)
- 21 System Test (FT)



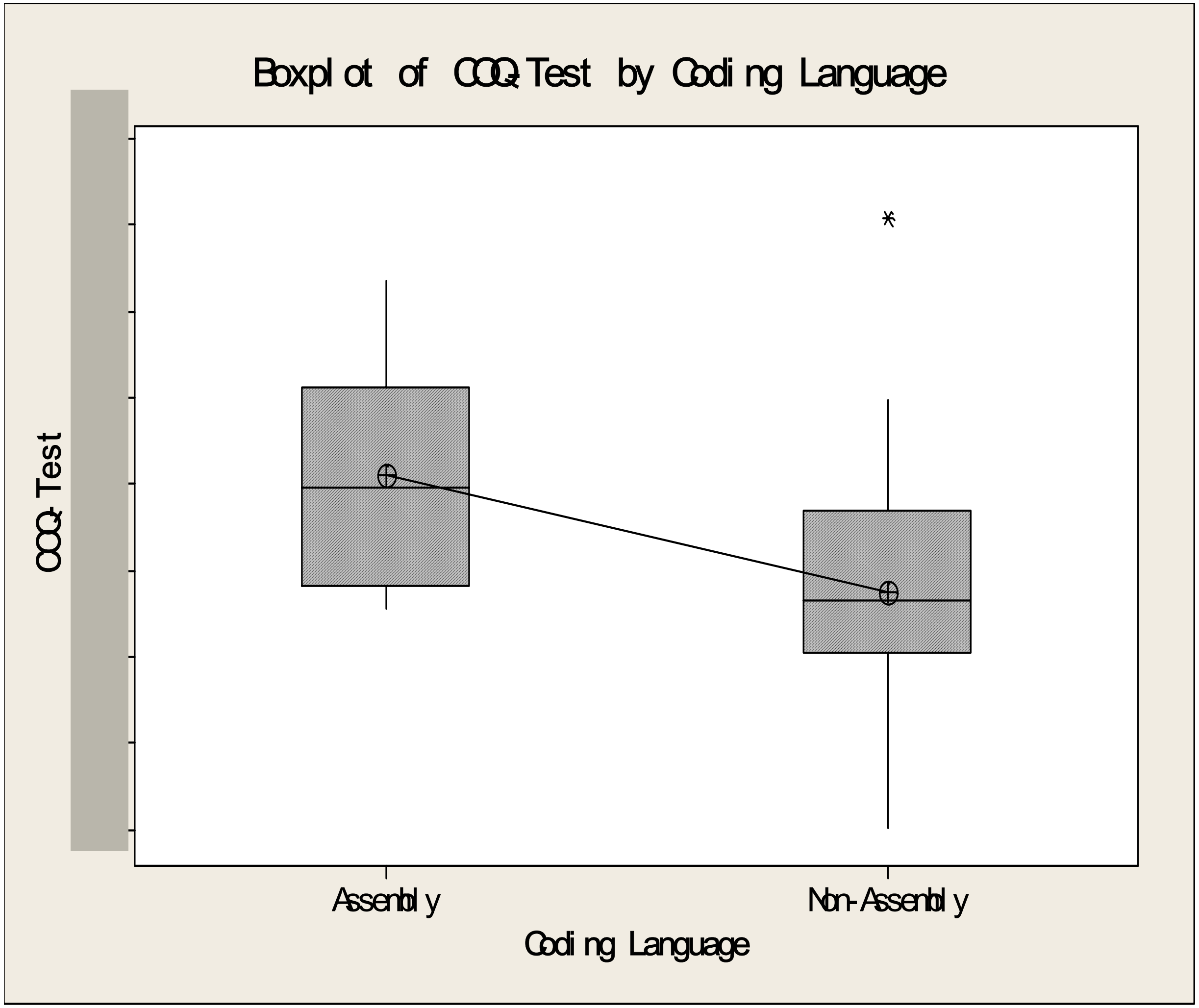
Differences by Coding Language

Major effects on test effort & COQ-Test

- Projects coded in assembly significantly more costly to test

Ns:

- 10 Assembly
- 26 Non assembly



Explaining the Variation

Factors that vary jointly with COQ-test – MANOVA Adjusted R^2 = .67

- Motorola end gates (Code & unit test; feature & component test; or system test)
- Delta Code Size in KLOC
- Domain (Multimedia, embedded, tools & telecom)
- Coding language (assembly versus non-assembly)
- Base Code size in thousand assembly-equivalent lines of code (KAELOC)

Other factors examined – weak relationships, not significant

- Project Lifecycle (full or partial)
- In Process Faults (IPF) ... Post Release Defects (PRD)
- COQ for review/inspection ... Total Document Size (pages)

Some Actions Taken

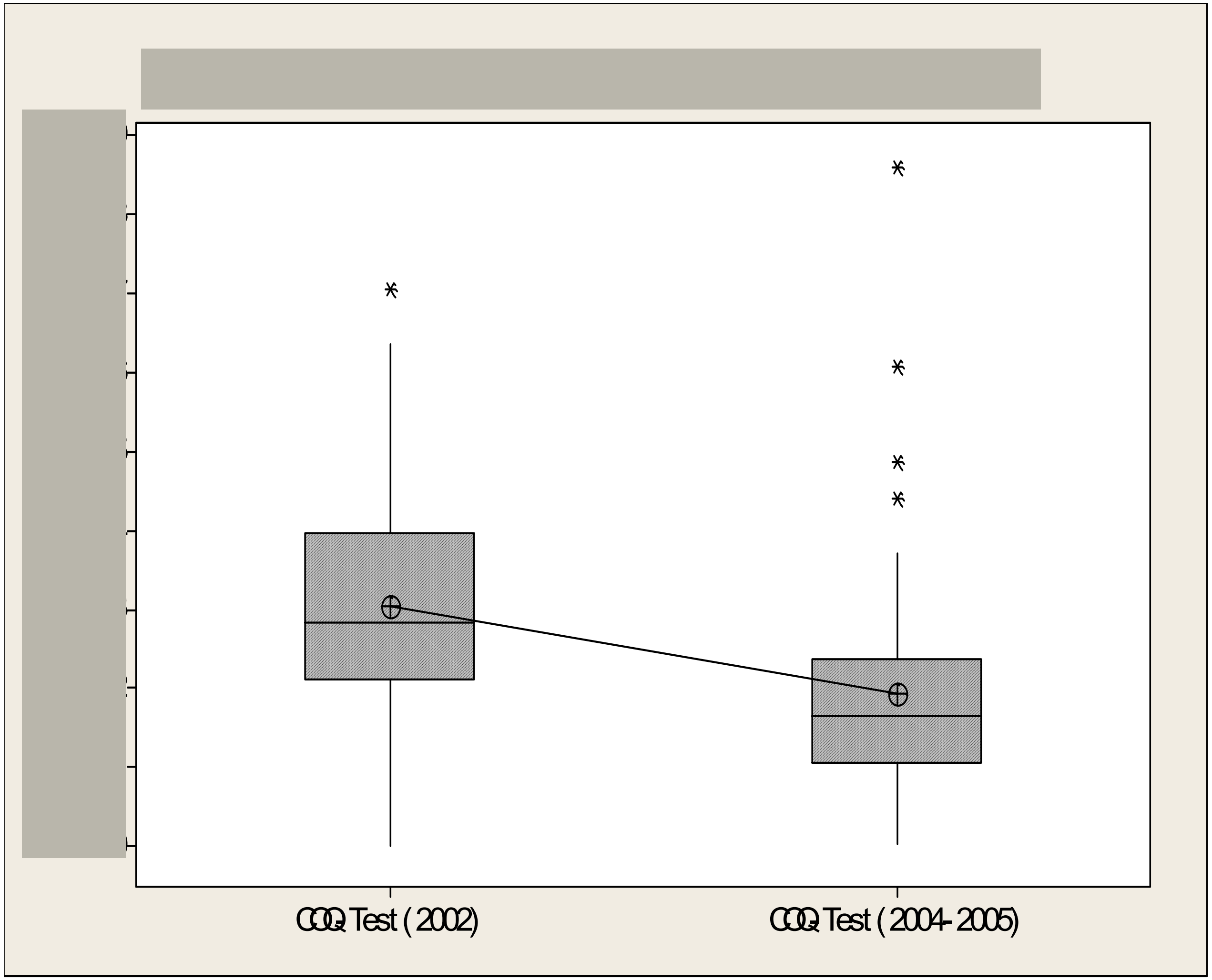
1. Encourage test automation
   - Especially for product-oriented projects
2. Develop organizational integration & system test guidelines
   - To reduce test development effort
3. Encourage project test case reuse & automation
   - Especially for long term projects
4. Enhance analysis of escaped defects
   - Develop causal analysis guidelines
   - Introduce causal analysis methods such as ODC
5. Optimize regression test strategy
   - Introduce fault prediction tool
6. Better sharing of practices & lessons learned among projects.

Effects of Process Change

Improvement actions reduced COQ-test cost

Ns:

- 56 in 2002
- 75 in 2004-2005



Phase Containment Effectiveness & In Process Faults at Motorola Software Group (MSG)

Phase Containment Effectiveness

- The proportion of faults that are found at the first check point after they were introduced
  - Classified by root cause analysis as errors ... as opposed to defects that escape detection until later
  - ***Examples shown here are for coding only***
- Updated incrementally by phase to monitor & control pertinent processes

In-Process Faults

- Number of faults found before completion of the project's final phase prior to release Customer Satisfaction

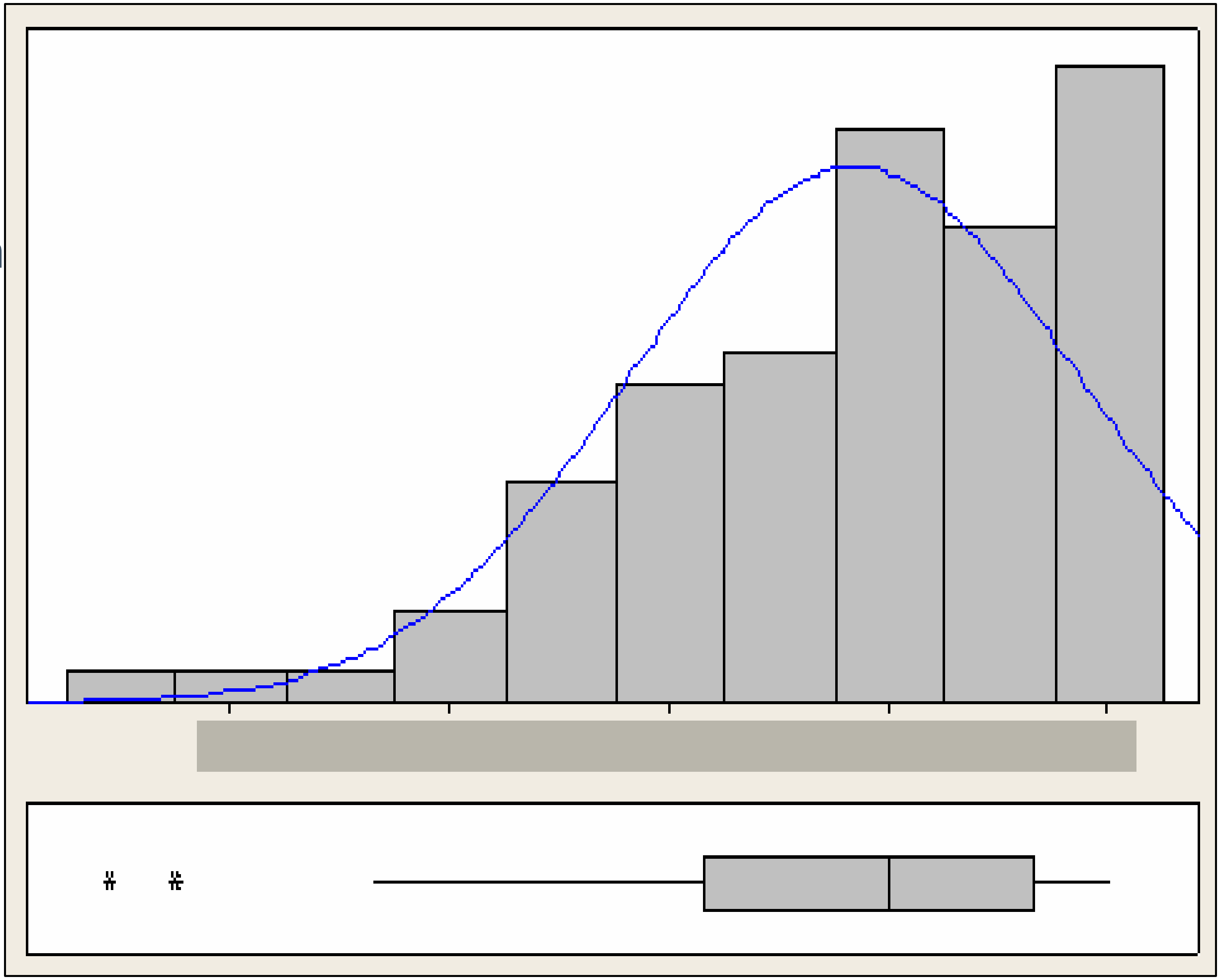
Related MSG performance measures (not shown here) include ...

- First Estimation Accuracy, Cycle Time Reduction Rate, Post-Release Defect Rate & Customer Satisfaction

Variation in Phase Containment Effectiveness

Results from MSG China Center

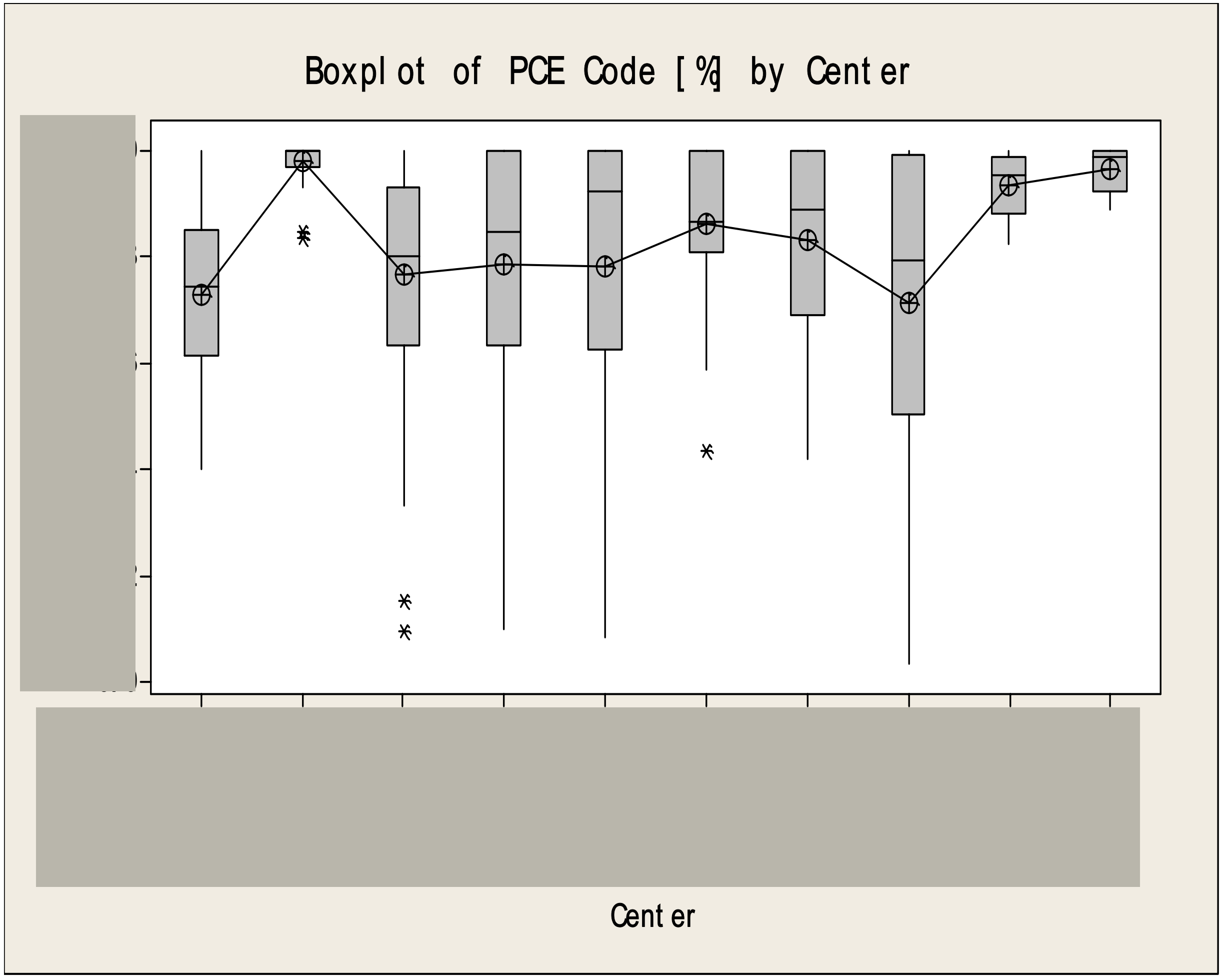
- N = 87; pooled from 2004 through 2006
- Actual values of data distribution (not shown here) are quite good by our sense of "industry standards"
- Still, there are variations that MSG wishes to reduce further



Phase Containment Effectiveness by MSG Center

Results from 10 MSG Centers

- N = 487
- All Centers do well by our sense of “industry standards”
- Again, there are variations that MSG wishes to reduce further



Phase Containment Effectiveness by Domain

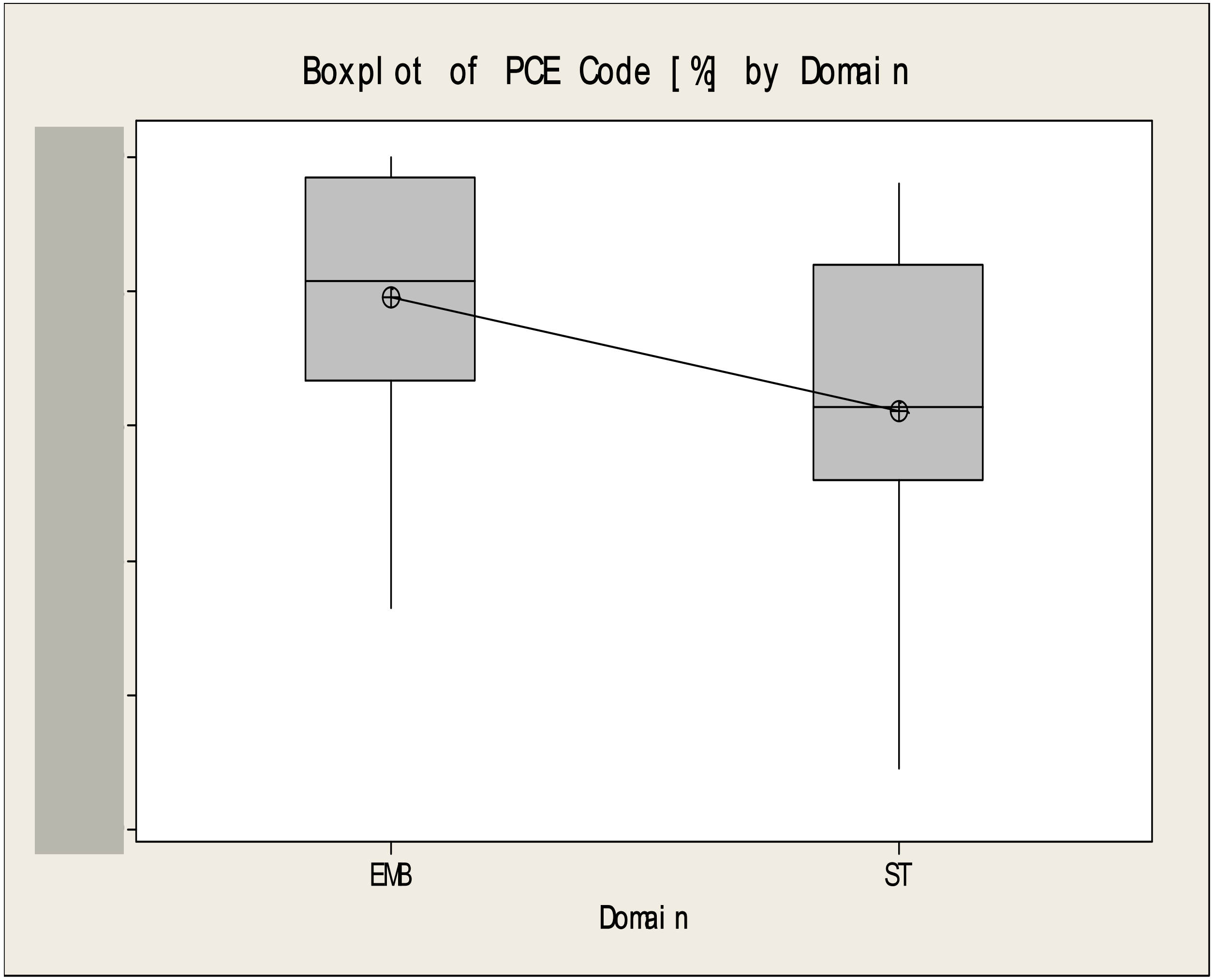
Better phase containment for Embedded projects

- Where China Center has more experience

Projects to develop Software Tools are newer work for the Center

Ns:

- 66 Embedded
- 15 Software Tools



Do Other Factors Affect Code Phase Containment Effectiveness?

Substantial differences among product lines

- Center does particularly well with multimedia projects – where they are very experienced

Yet, team experience has no independent overall effect

- Possibly implying adequate process definition & training

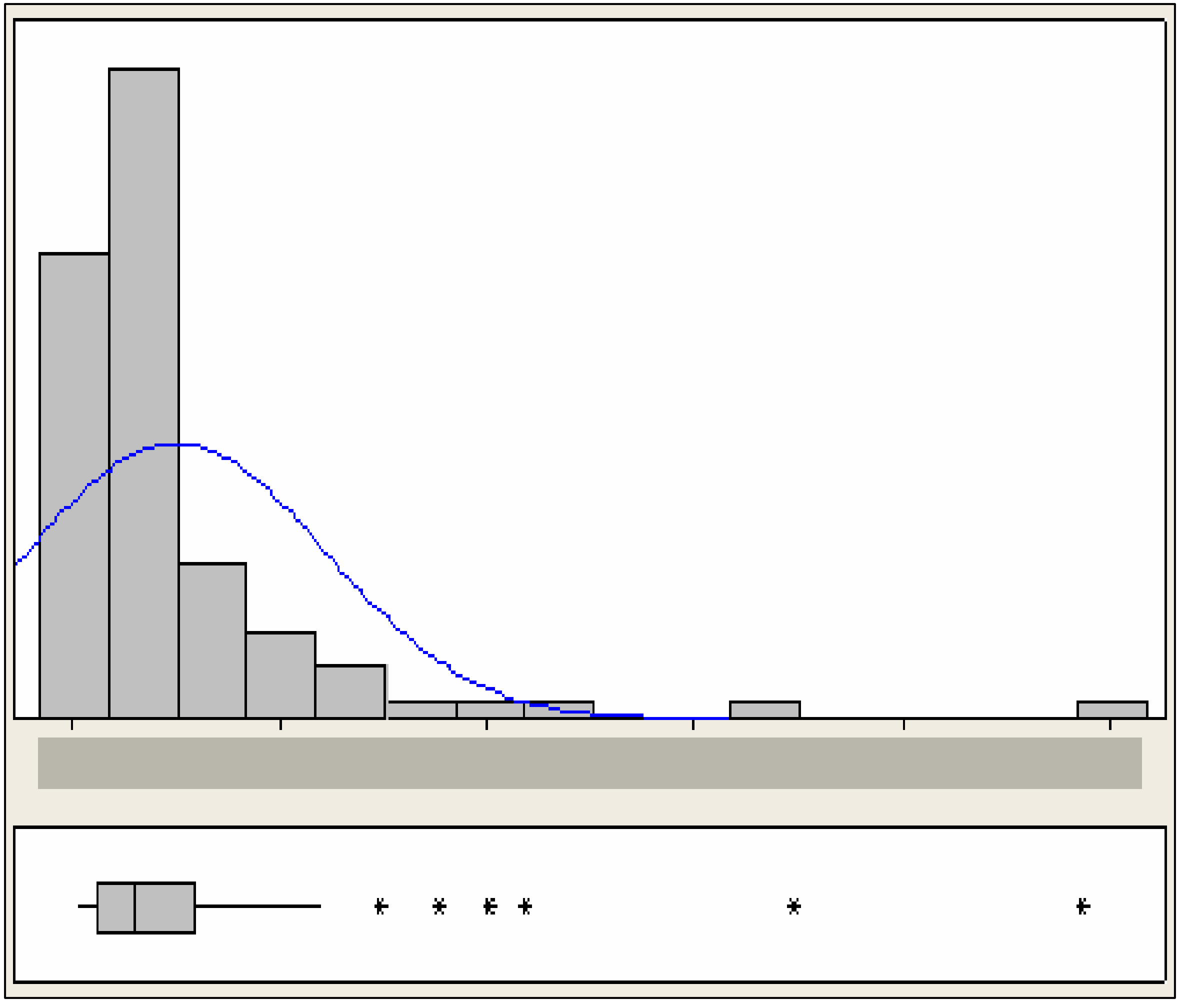
No significant differences by project category

- Customer completes some of the lifecycle
- Normal Development
- DMS
- Porting

Variation in In-Process Faults

Results from MSG China Center

- N = 87; pooled from 2004 through 2006
- Actual values of data distribution (not shown here) are quite good by our sense of “industry standards”
- Still, there are variations that MSG wishes to reduce further



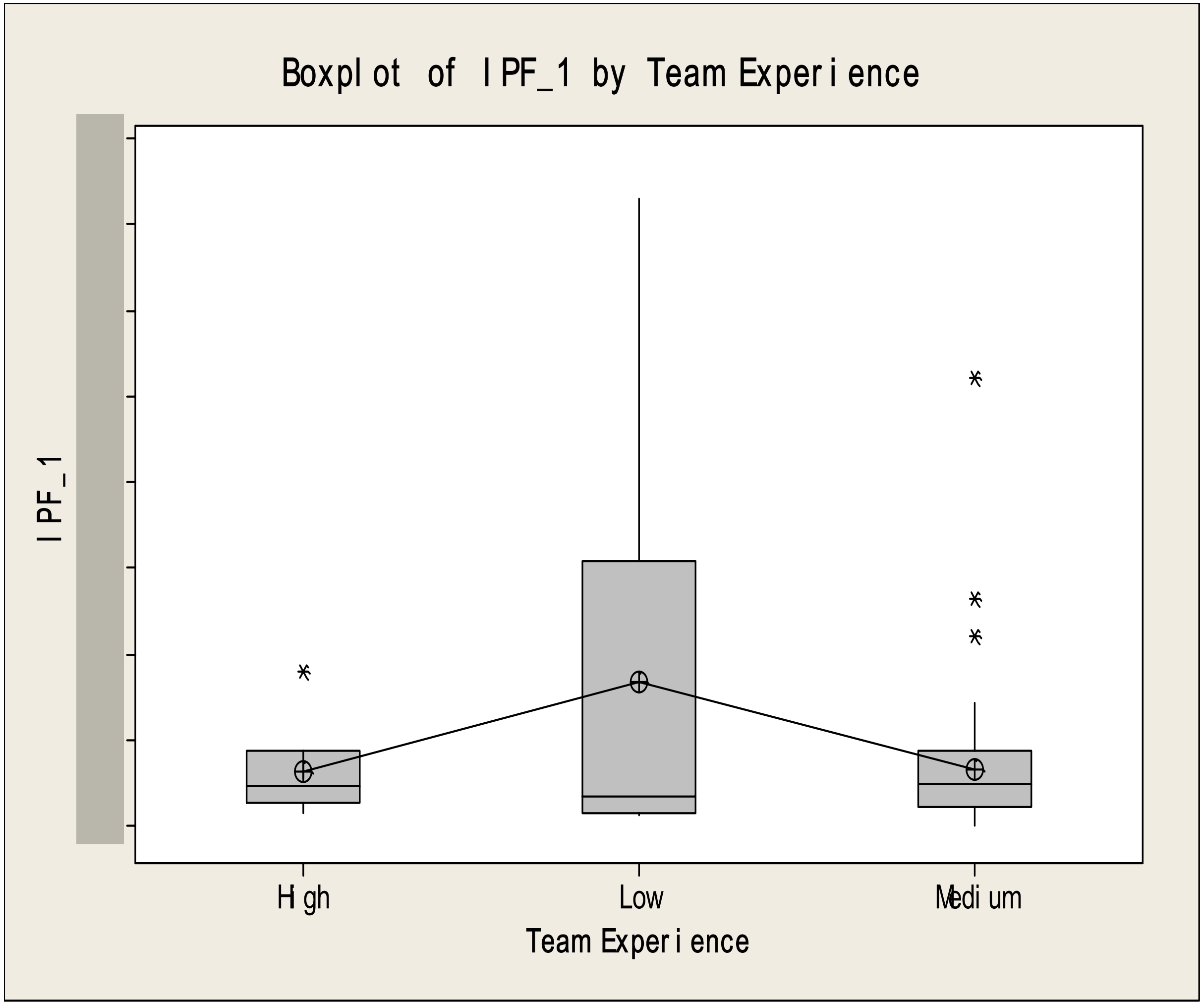
In-Process Faults by Team Experience

Team experience does affect insertion of faults

- Implying need for improved staffing, training, coding standards or other improved processes

Ns:

- 6 High
- 67 Medium
- 10 Low



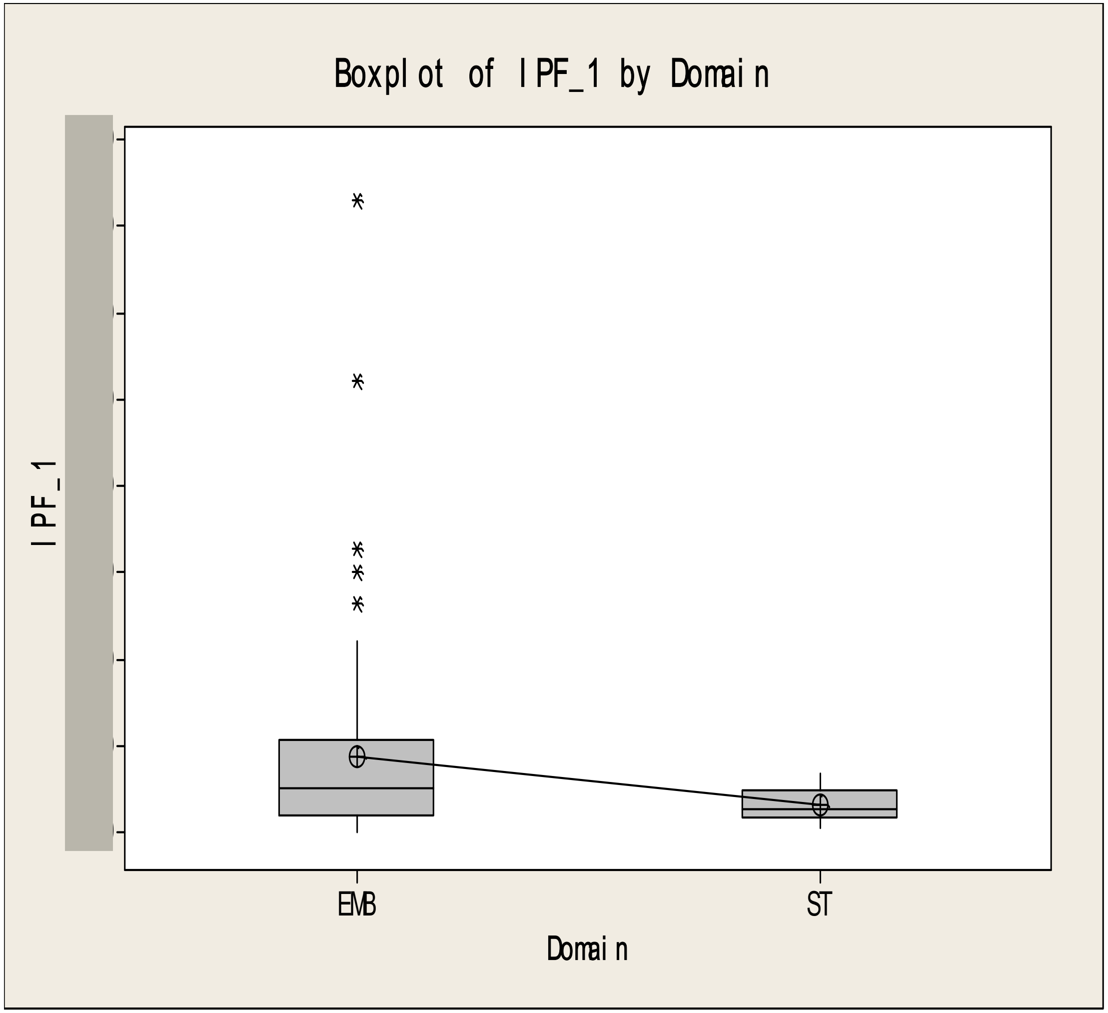
In-Process Faults by Domain

Marginally more faults in Embedded than Software Tools development projects

- Could be a function of greater complexity, imply need for improvement on engineering processes, or simply be statistical noise

Ns:

- 66 Embedded
- 15 Software Tools



Synopsis: Lockheed Martin Integrated Systems & Solutions

Performance Measures

- Unit Software Costs
- Find & fix costs
- Overhead rate
- Productivity (for equivalent lines of new code)
- Award fees

Processes instituted include

- Architecture based design
- Scenario based testing
- Design adequacy assessments
- Quality awareness throughout the lifecycle
- Early defect detection through inspections & defect causal analysis

Amassing the Necessary Evidence:
Some Related Examples

1. Independent measures of process compliance/adherence
2. CAR at Northrop Grumman IT
3. Project performance at Warner Robins Air Logistics Center

If short on time then punt ▶

Measuring Process Compliance & Adherence

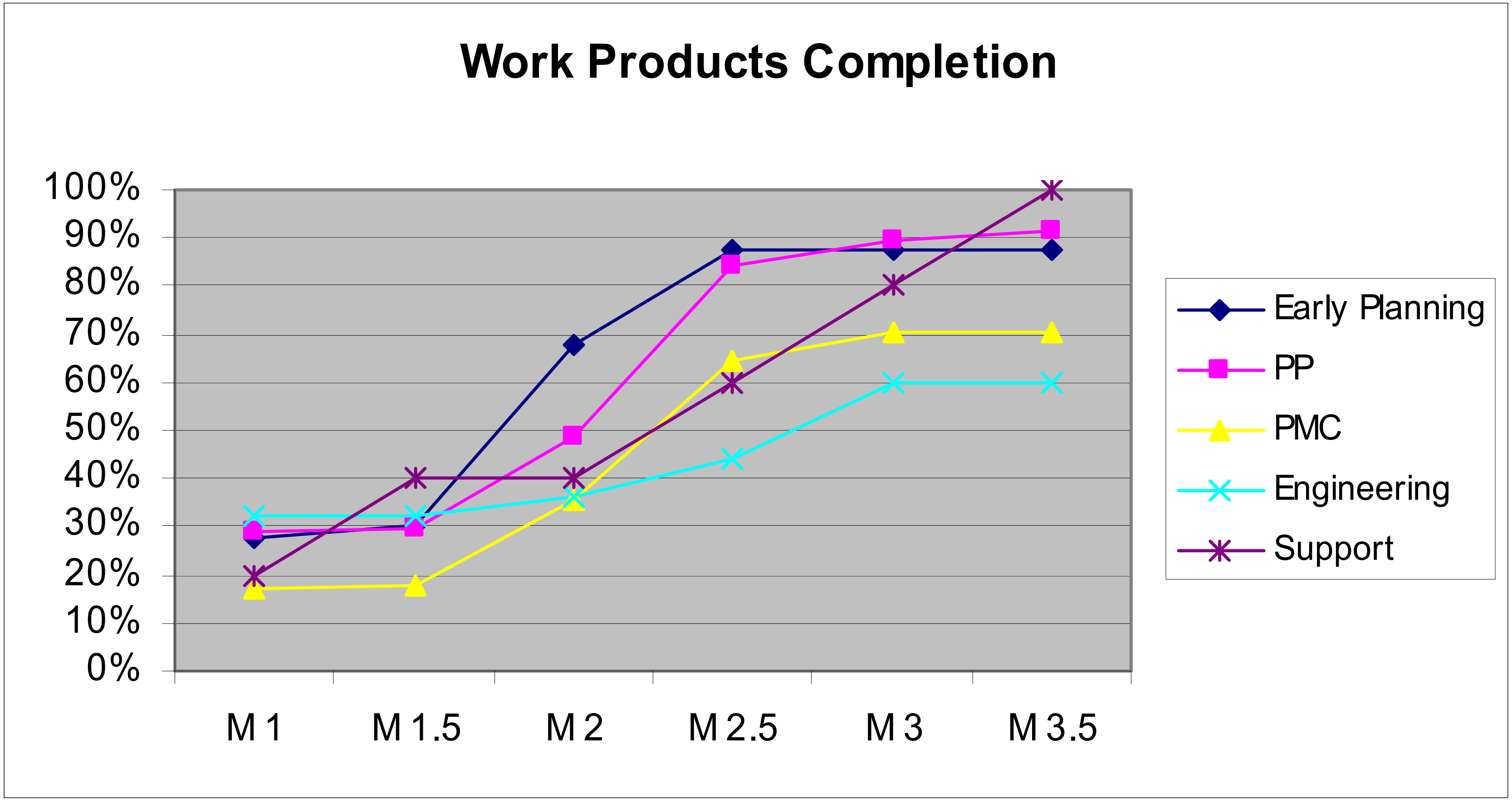
To do broadly-based comparisons

- Must already have (or begin to collect) independent measures of process capability

Available from

- CMMI class C appraisals
- Monitor & control activities
- QA tracking of existence, timeliness & quality of interim process work products:
- Questionnaires, e.g., in existing SW-CMM based studies

Measuring & Tracking Process Adherence



CAR at Northrop Grumman IT

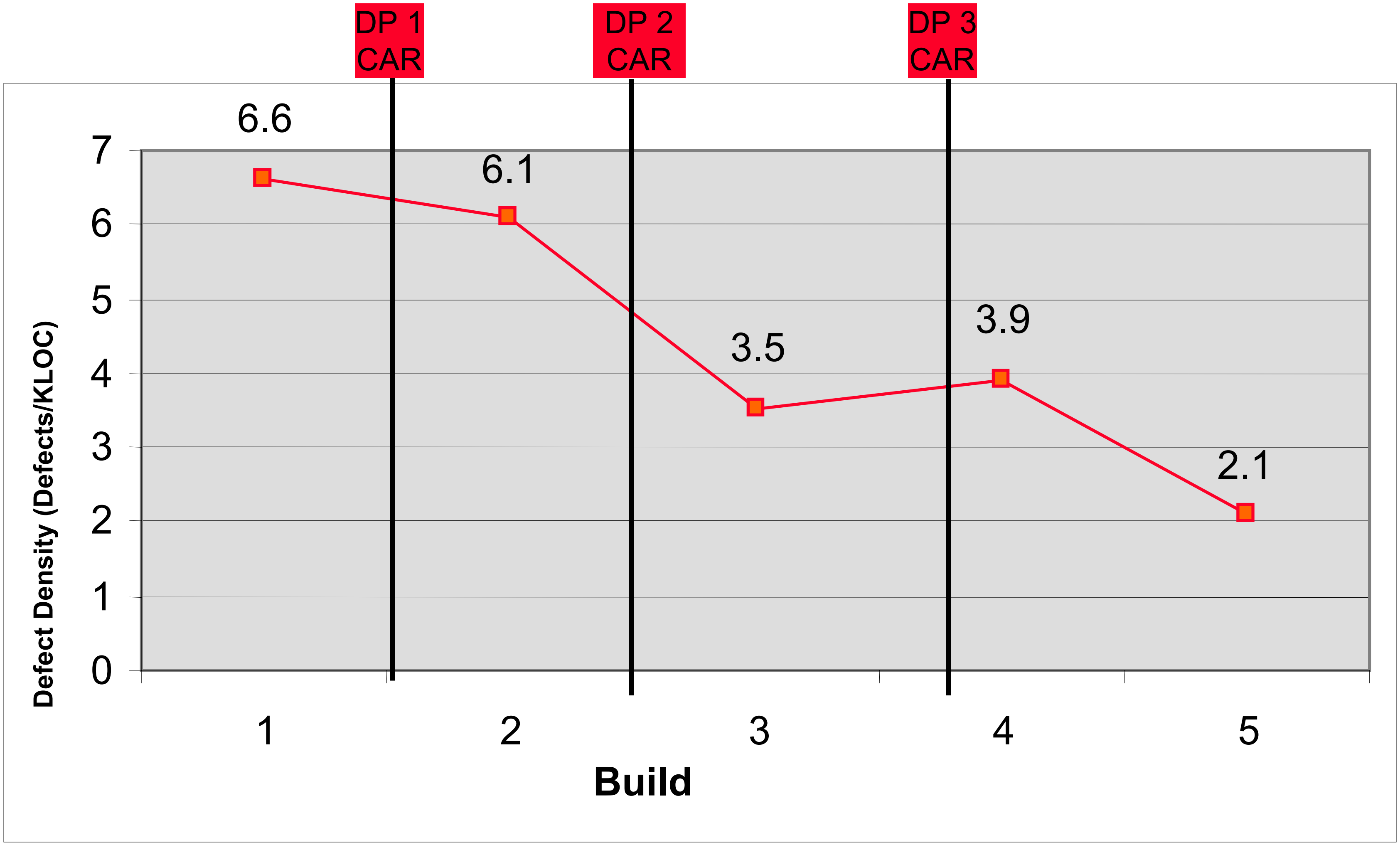
An example of careful CAR

- Albeit with one large project

Quasi-experimental control of change over time,

- Similar difficulty development tasks
- No other known differences over time
- A step jump after process intervention
  - Unlikely had the process change not taken place

CAR at Maturity Level 5: Northrop Grumman IT_1



CAR at Maturity Level 5: Northrop Grumman IT_2

Similar results for cost & schedule variance

ROI: ~13:1

- Hours saved fixing had defects not been avoided
- Hours invested on team training & conducting CAR Cycles

Project performance at Warner Robins Air Logistics Center

Comparisons of variation in performance over time

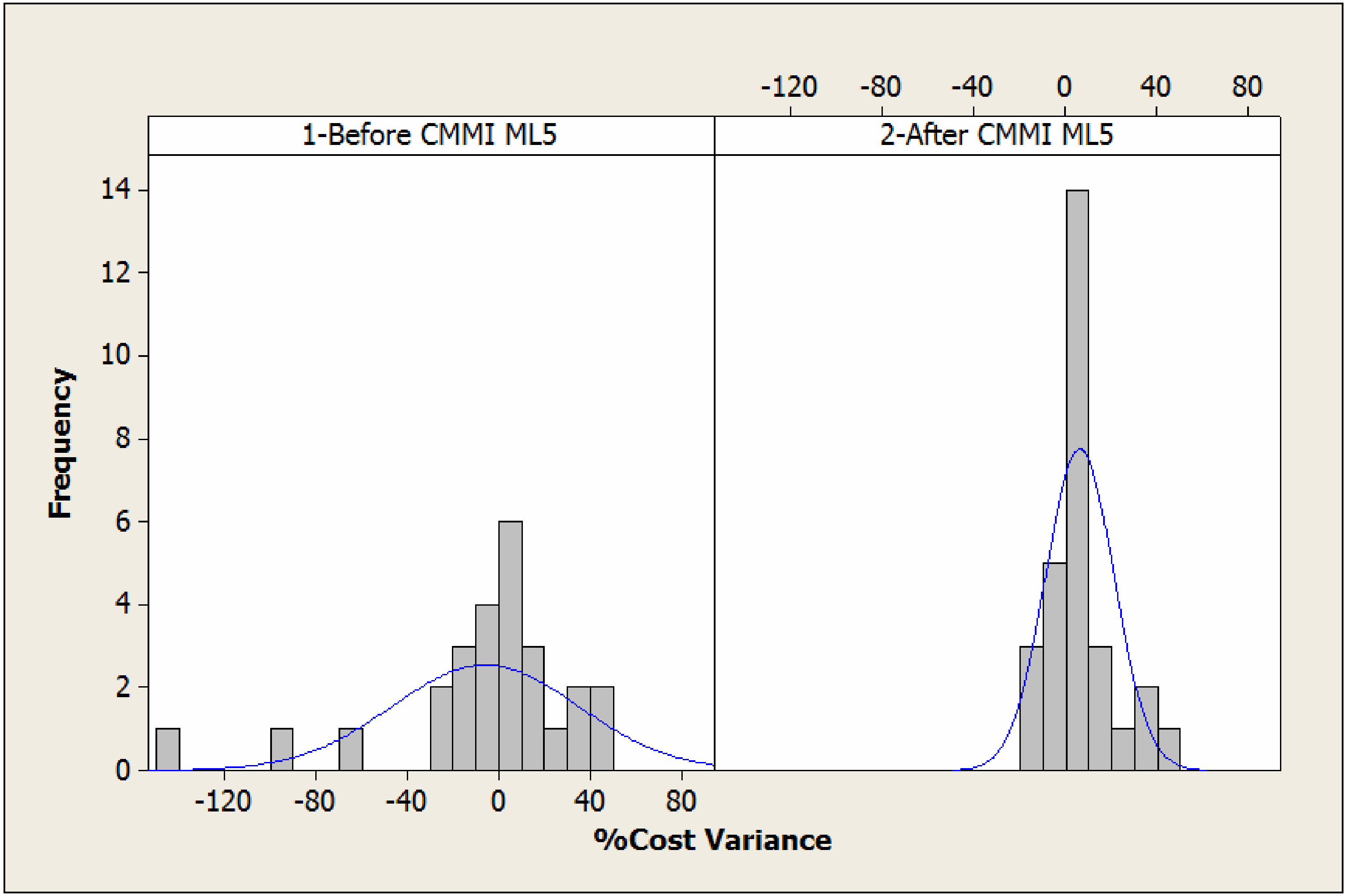
No explicit measurement of variation in process adherence by project

- But the process intervention clearly was the only thing that changed other than marginally

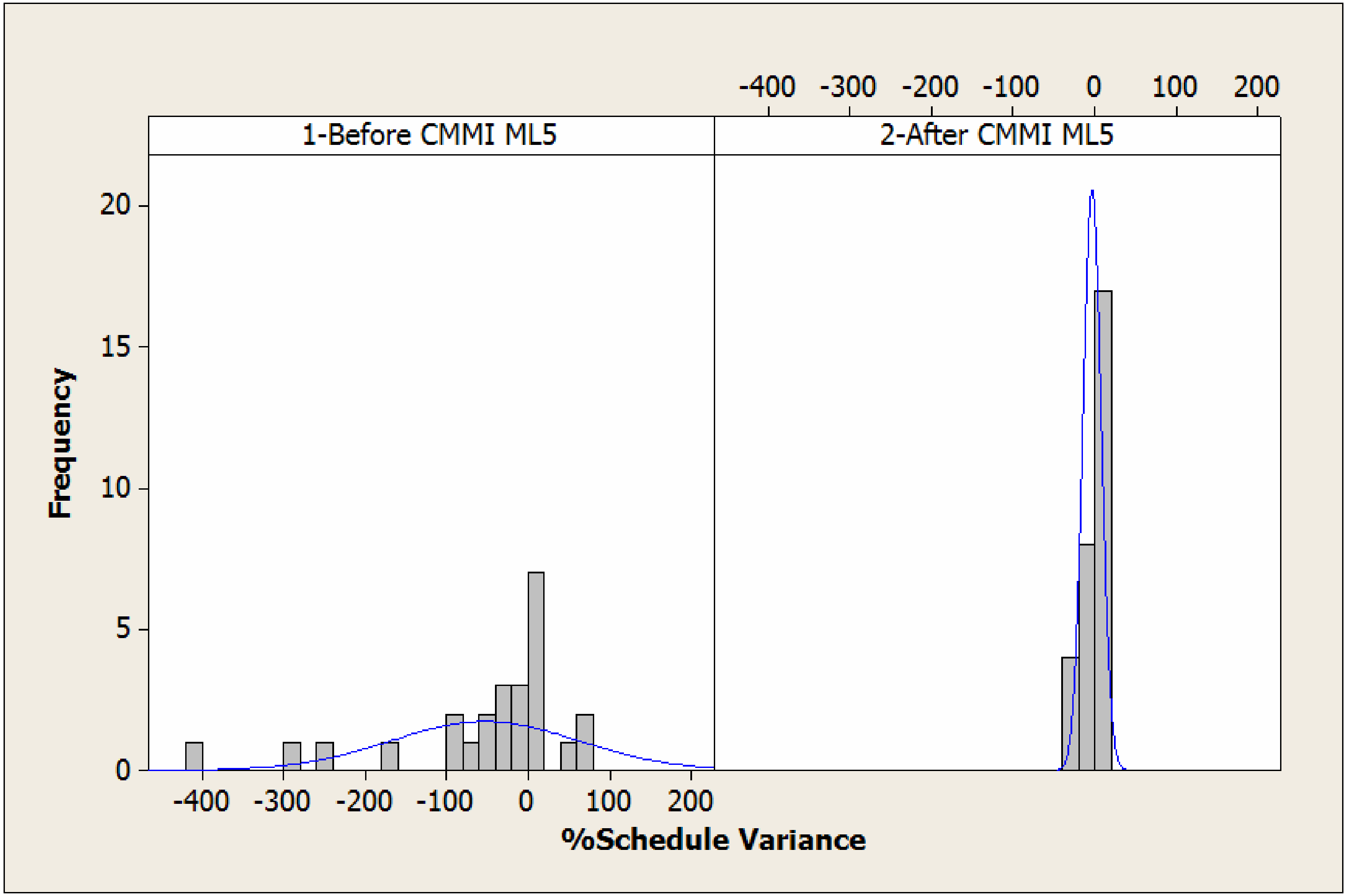
Small, short term maintenance projects

- Almost entirely adding new capabilities / functionality

Change in Percent Cost Variance



Change in Percent Schedule Variance



Today's Talk

Setting the stage

Our approach

Current & Proposed Studies

Enterprise wide performance benchmarking

Example results

How can you help?

How Can You Help?

Participate in Our Work & Contribute Your Own

We are always on the lookout for serious collaborators for:

- Enterprise Wide Performance Benchmarking studies
- The other work described today
- Any other ideas you may wish to pursue

Share your experiences with us & the wider systems& software community

- Submit a brief article for *DoD Software Tech News*, March 2007 Issue
  - Special issue on performance results of CMMI ... 31 January deadline
  - Quantitative case studies welcome ... comparisons of variations among projects/programs preferred

Also editing an issue (perhaps issues) of *Software Process Improvement and Practice* later in the year

Please see me here in Denver about these & other opportunities

Thank You for Your Attention!

Dennis R. Goldenson
dg@sei.cmu.edu

Software Engineering Institute
Carnegie Mellon University
Pittsburgh, PA 15213-3890
USA