



The Relative Cost of Interchanging, Adding, or Dropping Quality Practices

Bob McCann

Lockheed Martin Aeronautics

In developing systems and software, there are multiple opportunities to perform quality practices to find and to fix defects prior to putting the system or software into operations. This article demonstrates the following conclusions: 1) In general, quality practices should be ordered by increasing average cost to find and fix defects. Fixed costs do not affect this conclusion, but significant differences in either defect detection effectiveness or in the effectiveness of verifying rework induced defects can modify the conclusion. 2) One should retain or add a second quality practice provided the second practice fixes more defects during rework than the second practice creates during rework, provided the average cost to fix defects downstream is much larger than both the second practice's fixed costs and the second practice's marginal cost to find and fix defects.

In the beginning of a new project, project management gets to decide a very important issue, namely what work products are subject to a verification process, e.g., peer reviews, formal inspection, testing, etc. Some work products may even be deemed sufficiently critical that they are subjected to multiple verification processes during the development life cycle (e.g., requirements inspection, design inspection, code inspection, code desk check, compile and fix, informal peer reviews of various kinds, and various flavors of testing).

In these cases, there is nearly always a discussion of whether to use an informal peer review instead of a formal inspection and whether or not to skip the pre-compile desk check or to perform the code inspection before or after the first successful compile or even after the completion of unit testing. What is always present is the persistent nagging feeling that too much or too little was spent on verification. This article, together with the two previous articles on cost effectiveness of inspections¹, addresses that issue with a simple quantitative cost analysis model. It should be noted that this model is easily extended when a cause of variation in any of the factors become known.

In what follows, statistical reasoning is used (where that is not appropriate, the results may differ²). For instance, it is highly unlikely that a compilation test will discover a design defect (such as poor choice of algorithm) that results in a performance problem. In contrast, it is quite likely that a formal inspection of the design, a formal inspection of the code, and a formal performance test will all have a statistical likelihood of discovering that same design defect.

Warning: There may be simpler, more elegant proofs of the above results than what follows. If algebra or statistics give you a headache or other trauma, the

author apologizes for any discomfort from what follows.

Analysis

Suppose three or more adjacent quality practices that find and fix defects are performed in series (e.g., personal desk check, one-on-one peer review, compile and fix, formal inspection, unit testing, etc.³). Further suppose the cost effectiveness of each is measurable⁴ (please note that cost can be any independent variable of interest: dollars, labor hours, schedule impact, etc.):

- Let Q_j be quality practice j where j is 1, 2, ...
- Let F_j be the fixed and sunk costs of quality practice Q_j ⁵. Presumably F_j will be small compared to other cost terms if there are a significant number of defects.
- Let C_j be the average cost per defect found and fixed for practice Q_j including verification practice V_j .
- Let E_{Q_j} be the average effectiveness – fraction of defects present found for practice Q_j . Thus $E_{Q_j} * C_j$ would be the probable cost of finding and fixing a defect with quality practice Q_j .
- Let E_{V_j} be the average effectiveness – fraction of defects present found for practice V_j . Thus $E_{V_j} * C_j$ would be the probable cost of finding and fixing a defect with verification practice V_j .
- Let I_{R_j} be the number of defects inserted by the rework due to Q_j .
- Let I_0 be the number of defects inserted by earlier development practices. There is no breakage in this analysis if we only consider defects that are discovered sometime in the product life cycle. Defects that never get exercised have no actual impact, just potential impact.
- Let I_0 be the number of defects entering the second quality practice.
- Let III_{ij} be the number of defects

escaping both when Q_i precedes Q_j .

- Let R_j be the rework activity associated with quality practice Q_j .
- Let V_j be the average effectiveness of the verification of rework performed in Q_j . For algebraic simplicity we will assume V_j is approximately equal to E_j .
- Let T_{LC} be the total life cycle cost with subscripts indicating what quality practices are performed and in which order. Note that the letters *TLC* can also mean *Tender Loving Care* throughout the full life cycle, which is what this author thinks is necessary to comply with the intent of section 804 of the Bob Stump Act of 2003⁶. Thus, T_{LC12} refers to the case where Q_1 is performed before Q_2 .
- Let T_{LC12} be the total life-cycle cost when both Q_1 and Q_2 are present.
- Let T_{LC1} be the total life-cycle cost when Q_1 is present and Q_2 is absent.
- Let T_{LC2} be the total life-cycle cost when only Q_2 is present and Q_1 is absent.

Figure 1 shows the defect flow associated with quality practice Q_1 . In Figure 1 the circles represent tasks, and the boxes represent collections of defects.

- Q_1 is a quality practice that finds a fraction E_{Q1} of the defects present in some set of work products.
- R_1 is the task that does the impact analysis and repair for each of the identified defects. This in principle may introduce a new set I_{R1} of defects.
- V_1 is the verification task that follows the rework effort. It verifies the solutions to the defects discovered in Q_1 and finds a fraction E_{V1} of the new defects.
- Corrected defects do not propagate further.
- Escaped defects propagate to downstream processes.

When we stack two quality practices in

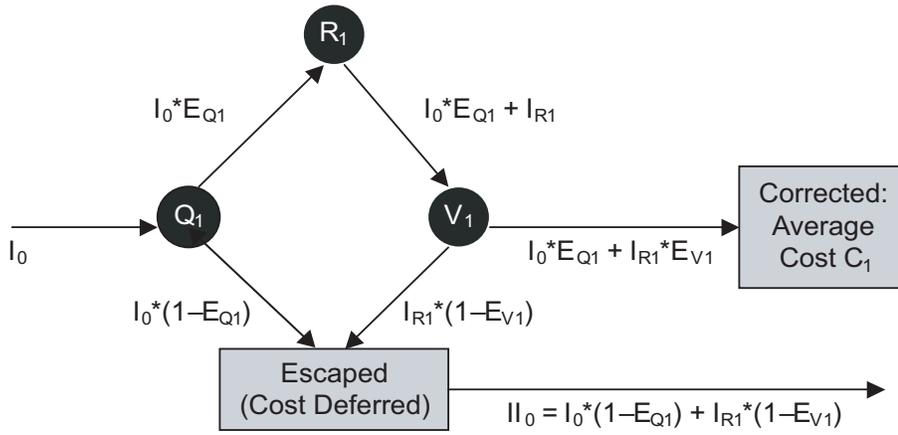


Figure 1: Defect Flow for Quality Practice Q_1

a row, the input to the second practice consists of the escaped defects from the first practice. Algebraically, this is accomplished by replicating Figure 1, changing the subscript 1 to 2 and replacing I_0 with $II_0 = I_0^*(1-E_{Q1}) + I_{R1}^*(1-E_{V1})$, as is shown in Figure 2.

Given the model described by Figure 1 and Figure 2, it is now algebraically possible to answer the following questions:

1. What is the cost increase in reversing the order of application of the two practices assuming all downstream defects eventually create an average cost C_3 per defect?

$$\begin{aligned}
 T_{LC\ 12} &= F_1 + C_1^*(I_{R1}^*E_{V1} + I_0^*E_{Q1}) + F_2 + C_2^* \\
 &\quad *(I_{R2}^*E_{V2} + II_0^*E_{Q2}) + F_3 + C_3^*III_{12} \\
 &= F_1 + C_1^*(I_{R1}^*E_{V1} + I_0^*E_{Q1}) \\
 &\quad + F_2 + C_2^* \{I_{R2}^*E_{V2} + [I_0^*(1-E_{Q1}) \\
 &\quad + I_{R1}^*(1-E_{V1})]^*E_{Q2}\} + F_3 + C_3^*III_{12} \\
 &= F_1 + F_2 + F_3 + C_1^*I_{R1}^*E_{V1} + C_2^*I_{R2}^*E_{V2} \\
 &\quad + C_1^*I_0^*E_{Q1} + C_2^*I_0^*(1-E_{Q1})^*E_{Q2} \\
 &\quad + C_2^*I_{R1}^*(1-E_{V1})^*E_{Q2} + C_3^*III_{12}
 \end{aligned}$$

$$\begin{aligned}
 T_{LC\ 21} &= F_1 + F_2 + F_3 + C_2^*I_{R2}^*E_{V2} \\
 &\quad + C_1^*I_{R1}^*E_{V1} + C_2^*I_0^*E_{Q2} \\
 &\quad + C_1^*I_0^*(1-E_{Q2})^*E_{Q1} + C_1^*I_{R2}^*(1-E_{V2}) \\
 &\quad *E_{Q1} + C_3^*III_{21}
 \end{aligned}$$

and

$$\begin{aligned}
 III_{12} &= I_{R2}^*(1-E_{V2}) + II_0^*(1-E_{Q2}) \\
 &= I_{R2}^*(1-E_{V2}) + [I_0^*(1-E_{Q1}) + I_{R1}^* \\
 &\quad *(1-E_{V1})]^*(1-E_{Q2}) \\
 III_{21} &= I_{R1}^*(1-E_{V1}) + [I_0^*(1-E_{Q2}) + I_{R2}^*(1-E_{V2})] \\
 &\quad *(1-E_{Q1})
 \end{aligned}$$

so

$$(III_{21} - III_{12}) = I_{R1}^*E_{Q2}^*(1-E_{V1}) - I_{R2}^*E_{Q1}^*(1-E_{V2})$$

thus

$$(T_{LC\ 21} - T_{LC\ 12}) = I_0^*(C_2^*E_{Q2} - C_1^*E_{Q1})$$

$$\begin{aligned}
 &+ I_0^*[C_1^*(1-E_{Q2})^*E_{Q1} - C_2^*(1-E_{Q1})^*E_{Q2}] \\
 &+ C_1^*I_{R2}^*(1-E_{V2})^*E_{Q1} - C_2^*I_{R1}^* \\
 &\quad *(1-E_{V1})^*E_{Q2} + C_3^*(III_{21} - III_{12}) \\
 &= I_0^*(C_2 - C_1)^*E_{Q1}^*E_{Q2} + C_1^*I_{R2}^* \\
 &\quad *(1-E_{V2})^*E_{Q1} - C_2^*I_{R1}^*(1-E_{V1}) \\
 &\quad *E_{Q2} + C_3^*(III_{21} - III_{12})
 \end{aligned}$$

$$\begin{aligned}
 (T_{LC\ 21} - T_{LC\ 12}) &= I_0^*(C_2 - C_1)^*E_{Q1}^*E_{Q2} \\
 &\quad + C_1^*I_{R2}^*(1-E_{V2})^*E_{Q1} - C_2^* \\
 &\quad *I_{R1}^*(1-E_{V1})^*E_{Q2} + C_3^*[I_{R1}^*E_{Q2} \\
 &\quad *(1-E_{V1}) - I_{R2}^*E_{Q1}^*(1-E_{V2})]
 \end{aligned}$$

Dividing this equation by $C_3^*I_0^*E_{Q1}^*E_{Q2}$ and setting the result to zero demonstrates the clarity of dimensionless ratios:

$$(T_{LC\ 21} - T_{LC\ 12}) / C_3^*I_0^*E_{Q1}^*E_{Q2} = 0$$

OR

$$\begin{aligned}
 (C_2 - C_1)/C_3 &+ [(C_3 - C_2)/C_3]^*(I_{R1}/I_0) \\
 &*(1-E_{V1})/E_{Q1} - [(C_3 - C_1)/C_3]^*(I_{R2}/I_0) \\
 &*(1-E_{V2})/E_{Q2} = 0
 \end{aligned}$$

$$(Equation\ 1)$$

The solution to this equation divides the parameter space into two regions – one in which the interchange is cost effective and one in which it is not. When the left-hand side is positive, it is more cost effective to perform Q_1 before Q_2 .

Although the first term is the one intuition would quickly identify, please note that because C_3 can be much larger than either C_1 or C_2 , the second and third terms may dominate the outcome, especially during the operations and maintenance phase. Note that the fixed cost contributions all cancel exactly.

In general, the quality practices should be ordered by increasing average cost to find and fix defects. Fixed costs do not affect this conclusion, but significant differences in either defect detection effectiveness or in the effectiveness of verify-

ing rework induced defects can modify the conclusion.

2. What is the cost of adding or dropping a quality practice?

A. Suppose we add or drop Q_1 :

$$\begin{aligned}
 T_{LC\ 12} &= F_1 + F_2 + C_1^*I_{R1}^*E_{V1} + C_2^*I_{R2}^* \\
 &\quad *E_{V2} + C_1^*I_0^*E_{Q1} + C_2^*I_0^*(1-E_{Q1})^*E_{Q2} \\
 &\quad + C_2^*I_{R1}^*(1-E_{V1})^*E_{Q2} + F_3 + C_3^*III_{12}
 \end{aligned}$$

where,

$$\begin{aligned}
 III_{12} &= I_{R2}^*(1-E_{V2}) + [I_0^*(1-E_{Q1}) \\
 &\quad + I_{R1}^*(1-E_{V1})]^*(1-E_{Q2})
 \end{aligned}$$

$$\begin{aligned}
 T_{LC\ 21} &= F_2 + C_2^*(I_{R2}^*E_{V2} + I_0^*E_{Q2}) \\
 &\quad + F_3 + C_3^*III_{21}
 \end{aligned}$$

where,

$$III_{21} = I_0^*(1-E_{Q2}) + I_{R2}^*(1-E_{V2})$$

so

$$(III_{21} - III_{12}) = [I_0^*E_{Q1} - I_{R1}^*(1-E_{V1})]^*(1-E_{Q2})$$

thus

$$\begin{aligned}
 (T_{LC\ 21} - T_{LC\ 12}) &= (I_0^*E_{Q1} + I_{R1}^*E_{V1})^*(C_2^*E_{Q2} - C_1) \\
 &\quad + C_3^*[I_0^*E_{Q1} - I_{R1}^*(1-E_{V1})]^*(1-E_{Q2}) \\
 &\quad - [F_1 + C_2^*I_{R1}^*E_{Q2}]
 \end{aligned}$$

This case will require individual analysis using actual (or accurately estimated) cost performance data.

Keeping/adding Q_1 is better if

$$\begin{aligned}
 (I_0^*E_{Q1} + I_{R1}^*E_{V1})^*(C_2^*E_{Q2} - C_1) \\
 + C_3^*[I_0^*E_{Q1} - I_{R1}^*(1-E_{V1})]^*(1-E_{Q2}) \\
 > [F_1 + C_2^*I_{R1}^*E_{Q2}]
 \end{aligned}$$

$$(Equation\ 2)$$

Indeed, if C_3 is sufficiently large and I_{R1} is sufficiently small, it will always be practical to keep/add a quality practice. However, if I_{R1} is sufficiently large, then the converse will be true. In this case, the cost incurred due to mistakes inserted during rework swamps the value of mistakes actually found and fixed. Under those conditions it is better to drop the (broken) quality practice.

It is also true that very high fixed costs can cause a quality practice to become impractical. This is especially true when the *cost* of primary concern is a very aggressive development schedule commitment. It takes serious discipline on the part of both development management and customer management to put long term goals before short term concerns. Recent congressional and Department of Defense efforts to emphasize total life-cycle costs appears to be an attempt to

provide a context in which this long term focus is even possible⁷.

B. Suppose we add or drop Q₂:

$$T_{LC1} = F_1 + C_1*(I_{R1}*E_{V1} + I_0*E_{Q1}) + F_3 + C_3*III_1$$

$$T_{LC12} = F_1 + C_1*(I_{R1}*E_{V1} + I_0*E_{Q1}) + F_2 + C_2*(I_{R2}*E_{V2} + I_0*E_{Q2}) + F_3 + C_3*III_{12}$$

$$(T_{LC1} - T_{LC12}) = C_3*(III_1 - III_{12}) - F_2 - C_2*(I_{R2}*E_{V2} + I_0*E_{Q2})$$

but $III_1 = I_0$

so

$$(T_{LC1} - T_{LC12}) = C_3*(I_0 - [I_0*(1 - E_{Q2}) + I_{R2}*(1 - E_{V2})]) - F_2 - C_2*(I_{R2}*E_{V2} + I_0*E_{Q2})$$

$$= C_3*(I_0*E_{Q2} - I_{R2}*(1 - E_{V2})) - C_2*(I_{R2}*E_{V2} + I_0*E_{Q2}) - F_2$$

$$= (C_3 - C_2)*I_0*E_{Q2} + (C_3 - C_2)*I_{R2}*E_{V2} - C_2*I_{R2}$$

We should retain/add Q₂ provided $(T_{LC1} - T_{LC12}) > 0$. This can be expressed as the following:

$$(C_3 - C_2)*I_0*E_{Q2} + (C_3 - C_2)*I_{R2}*E_{V2} > F_2 + C_2*I_{R2}$$

or

$$I_0*E_{Q2} + I_{R2}*E_{V2} > (C_2*I_{R2} + F_2)/(C_3 - C_2)$$

(Equation 3)

Therefore, one should retain/add Q₂ provided the second practice fixes more defects during rework than the second practice creates during rework provided C₃ is much larger than either F₂ or C₂.

Worked Example

To make the results more solid, consider a software development effort delivering a million lines of code over five years by a team of 100 software developers. Given that software developers tend to change jobs quickly to keep their skills current, one can assume that defects found during design and coding will be fixed and verified by the original author and that defects found late in testing will be fixed and verified by someone other than the original author. Fixed and sunk costs will be ignored and some average error rates and costs for this team of developers will be guessed:

- Twenty defects per thousand lines of code inserted during coding and design: I₀ = 20,000.
- Inspection catches three out of every our defects present: E_{Q1} = 0.75.
- Three new defects are created for every 10 fixed: I_{R1} = 0.3*0.75*20,000 =

4,500 (please note this assumes I_{R1} is proportional to I₀).

- Defect detection and repair costs about one labor hour each: C₁ = 1.0.
- Rework detection catches nine out of 10 newly created defects: E_{V1} = 0.9.
- Pre-delivery testing detects six defects out of 10 defects: E_{Q2} = 0.6.
- Test rework inserts five new defects for every 10 fixed (not the original author): I_{R2} = 1,635.
- Test rework detection catches six out of 10 newly created defects (not the original rework verifier): E_{V2} = 0.6.
- Test detection and repair costs 40 labor hours each: C₂ = 40.
- Post delivery error detection and repair costs 100 labor hours each: C₃ = 100.
- Ignore fixed and sunk costs: F₁ = F₂ = F₃ = 0.

In this case, spreadsheet analysis can be used to compute the various costs in labor hours:

- T_{LC21} - T_{LC12} = 318,614 > 0 (do not permute the inspection and testing!).
- T_{LC1} - T_{LC12} = 91,560 > 0 (do not drop the inspection).
- T_{LC,2} - T_{LC12} = 912,150 > 0 (do not drop the testing).

In this case, permuting the practices raises costs, as does dropping either practice.

Just for fun, it is now possible to guess what happens when the two practices are inspection and (Q₁) unit test (Q₂). Which should one do first? Assume unit tests are 90 percent effective at finding defects, but take four hours each to find and fix the defect (additional unit tests get custom-built to diagnose and localize the defects) and verification finds 60 percent of defects created by the rework:

- T_{LC21} - T_{LC12} = 30,821 > 0 (do the inspection first).
- T_{LC1} - T_{LC12} = 401,556 > 0 (do not drop the inspection).
- T_{LC,2} - T_{LC12} = 178,830 > 0 (do not

drop the testing).

This was assuming unit tests were 90 percent effective. One can ask at what unit test effectiveness the two practices are neutral to interchange; in this case, approximately 55 percent. In this case, if the unit tests are less than 55 percent effective at detecting defects, they should be performed prior to the inspection.

Note: Please do not quote these example results! Plug in your own measurements and get real answers to your questions.

Indeed, if we use Equation 1, we can algebraically solve for interchange neutrality. On a diagram of the parameter space, the solution to this equation would divide the space into two regions, one in which the interchange is cost effective and one in which it is not:

$$(C_2 - C_1)/C_3 + [(C_3 - C_2)/C_3]*(I_{R1}/I_0)*(1 - E_{V1})/E_{Q1} - [(C_3 - C_1)/C_3]*(I_{R2}/I_0)*(1 - E_{V2})/E_{Q2} = 0$$

This can be solved for 1/E_{Q2} directly:

$$1/E_{Q2} = \{I_0/[(1 - E_{V2})*I_{R2}]*[(C_2 - C_1)/(C_3 - C_1) + [(C_3 - C_2)/(C_3 - C_1)]*(I_{R1}/I_0)*(1 - E_{V1})/E_{Q1}]\}$$

(Equation 4)

provided the following inequality constraint holds:

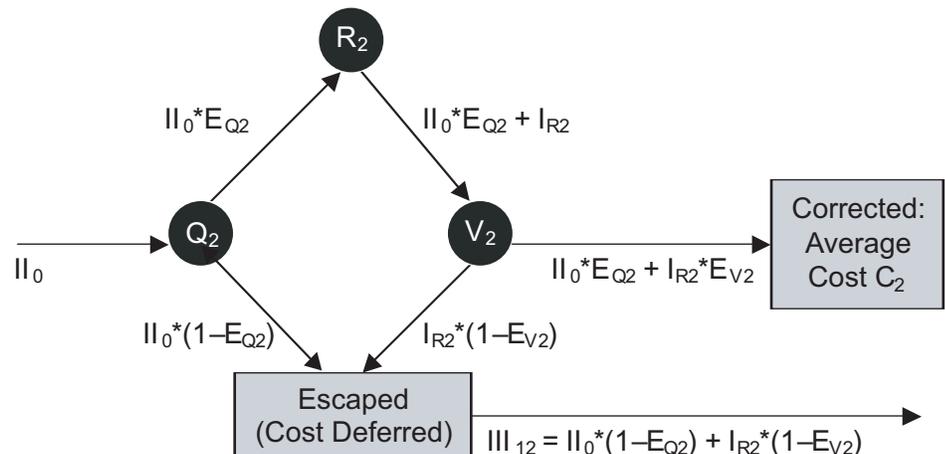
$$0 < E_{Q2} \leq 1$$

Conclusions

This analysis has demonstrated the following conclusions:

- In general, the quality practices should be ordered by increasing average cost to find and fix defects. Fixed costs do not affect this conclusion, but significant differences in either defect detection effectiveness or in the effective-

Figure 2: Downstream Defect Handling



ness of verifying rework induced defects can modify the conclusion.

- One should retain/add Q_2 provided the second practice fixes more defects during rework than the second practice creates during rework provided C_3 is much larger than either F_2 or C_2 .

The Measurement and Traceability Challenge

Although the above analysis does not appeal to counter-intuitive reasoning as is sometimes the case with statistical reasoning, there is a much more demanding barrier to benefiting from this analysis: getting organizations to track defects to their origin and to measure the associated costs of finding and fixing them. One problem is that quality practices are not always performed and measured the same way. Nor do they necessarily define or count defects in the same way.

To utilize the analysis presented here, each quality practice would need to measure the following items:

- C : The average cost to find and fix a defect discovered during the practice.
- I_0 : The number of defects inserted prior to the practice.
 - Either exclude those that are never found at all during the product life cycle – they carry no actual cost, just potential cost.
 - Or, use a fault injection based experimental design to estimate I_0 with known accuracy⁸.
- I_R : The number of defects inserted during rework resulting from the practice.
- E_Q : The fraction of incoming defects (I_0) found by the quality practice.
- E_V : The fraction of defects inserted during rework (I_R) found during verification.

Even though there are only five items to measure, it is necessary that all quality practices use the same defect definition and that all⁹ defects get traced to their point of insertion, preferably by an automated process. Effective version control and configuration management are essential here. Further, the variable costs, sunk costs, and fixed costs used in finding and fixing the defects would need to be captured. ♦

Acknowledgements

The author is indebted to Dr. Charles Farmer and Lynn Rabideau for making the time available to write the article and to his wife Carmen Lopez for tolerating a persistently scientific approach to life's common problems.

⁸ CMMI is registered in the U.S. Patent and Trademark Office by Carnegie Mellon University.

Notes

1. See McCann. "How Much Code Inspection Is Enough?" *CROSSTALK* July 2001, and "When Is It Cost Effective to Use Formal Software Inspections?" *CROSSTALK* Mar. 2004.
2. For this analysis to be valid, it is necessary to have enough data that the concepts of *confidence interval* and *hypothesis testing* are well defined for the quantities of interest, typically mean and standard deviation. In the case of single humped distributions, required sample size is proportional to the standard deviation of the data. Statistically stable practices have less variation than statistically unstable practices, so they require less data to reach valid conclusions. The exact number of data points depends on the specific distribution being used. Analyses that is free of distribution assumptions (non-parametric analysis) typically take more, not less data.
3. The case of three adjacent practices is sufficient. The general case can be derived using the same analytic approach, although with a bit more algebraic effort. Please note that if two practices find orthogonal sets of defects, then permuting them has no effect on overall cost effectiveness.
4. Measurability of various things will depend on the process maturity of the organization. Capability Maturity Model Integration (CMMI[®]) Level 5 organizations will routinely address information needs related to process cost effectiveness. CMMI Level 1 organizations will be much less likely to be able to do so.
5. A fixed cost is one which does not grow in proportion to activity performed, see <http://en.wikipedia.org/wiki/Fixed_cost>. A sunk cost is one which has already been incurred and which cannot be recovered to a significant degree, see <http://en.wikipedia.org/wiki/Sunk_cost>. Although they are quite different, those differences are not relevant to this analysis.
6. See Section 804 <www.dod.mil/dodgc/olc/docs/2003NDAA.pdf#search=%22bob%20stump%20act%20of%202003%22>.
7. See for instance (current as of 8/2006):
 - <http://lean.mit.edu/index.php?option=com_content&task=view&id=47&Itemid=57>.
 - <www.dau.mil/conferences/2005/Wednesday/B1-1345-McElroy-CR73.pdf>.
 - <www1.eere.energy.gov/femp/pdfs/lcc_guide_05.pdf>.
8. Executive Orders 13101 and 13123.
9. OMB Circular A-94.
 - FAR Part 7.1, especially 7.105 and Part 52.248-2 (b) and 52.248-3.
10. Defense Acquisition Guidebook, section 5.1.3.5, "Life Cycle Cost Optimization" and Chapter 3 "Affordability and Life-Cycle Resource Estimates."
11. We can put a lower bound on the probable downstream cost of undiscovered defects actually discovered later in the life cycle. Given defect injection techniques, it is actually possible to get statistically valid estimates of the number of undiscovered defects. This technique is discussed thoroughly in Mills, Harlan D. "Statistical Validation of Computer Programs" and in "Software Productivity." Dover House Publishing, 1988.
12. If a program collects enough data and has a repeatable practice of using statistical techniques in process management, then *all* can be relaxed to the idea of a *statistically significant sample*.

About the Author



Bob McCann is a staff systems engineer at Lockheed Martin Aeronautics in Fort Worth, Texas. He has nearly 20 years of experience in computational physics and high performance computing, including nine years at Princeton Plasma Physics Laboratory working in the U.S. Department of Energy-controlled fusion program. McCann has served as a member of the Lockheed Martin Integrated Systems and Solution Metrics Process Steering Committee and currently works on improving systems engineering processes, methods, and metrics. He has a bachelor's degree and a master's degree in physics, a master's degree in computer science, and a master's degree in computer systems management/software development management.

Lockheed Martin Aeronautics
P.O. Box 748, MZ 8605, BLDG F500
Fort Worth, TX 76101
Phone: (817) 935-4037
Fax: (817) 935-5272
E-mail: bob.mccann@lmco.com