Statistical Tune-Up of the Peer Review Engine to Reduce Escapes

Tom Lienhard, Raytheon Missile Systems

Abstract. Peer reviews are a cornerstone to the product development process. They are performed to discover defects early in the lifecycle when they are less costly to fix. The theory is to detect the defects as close to the injection point as possible reducing the cost and schedule impact. Like most, if not all companies, peer reviews were performed and data collected allowing characterization of those reviews. Data collected across the organization showed that more than 30% of the engineering effort was consumed by reworking products already deemed fit for purpose. That meant for every three engineers a fourth was hired just to rework the defects. This was unacceptable!

The major contributor to this rework was defects that escaped or "leaked" from one development phase to a later phase. In other words, the peer reviews were not detecting defects in the phase during which they were injected. Defect leakage is calculated as a percentage, by summing the defects attributable to a development phase that are detected in later phases divided by the total number of defects attributable to that phase. Defect leakage leads to cost and budget over-runs due to excessive

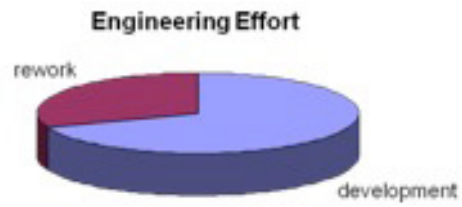


Figure 1

rework. For some development phases, defect leakage was as high as 75%. By investigating the types of defects that go undetected during the various development phases, corrections can be introduced into the processes to help minimize defect leakage and improve cost and schedule performance. An organizational goal was then set at no more than 20% defect leakage.

To perform this investigation and propose improvements, a suite of Six Sigma tools were used to statistically tune-up the peer review process. These tools included Thought Process map, Process Map, Failure Mode and Effect Analysis, Product Scorecard, Statistical Characterization of Data, and a Design of Experiments.

Having been an engineer and process professional for more than 20 years, I knew (or thought I knew) what influenced the peer review process and what needed to be changed in the process. But when we began the process, I kept an open mind and used Six Sigma tools to characterize and optimize the peer review process.

The Thought Process Map was needed to scope the project, keep the project on track, identify barriers, and document results. It was useful to organize progress and eliminate scope-creep.

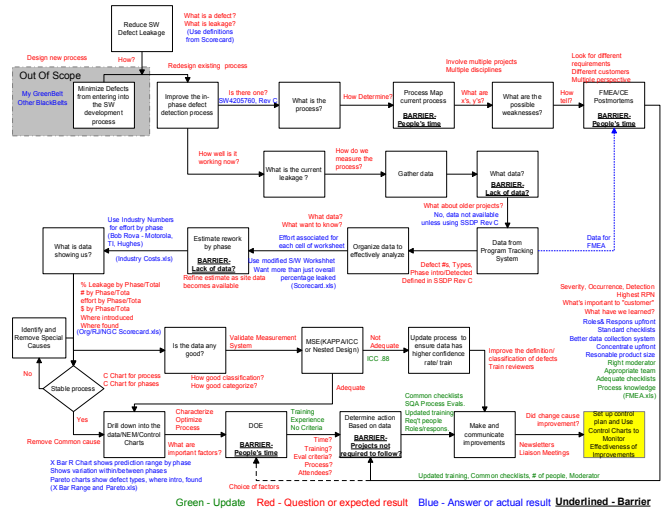


Figure 2

The Process Map was used to "walk the process" as it is implemented—not as it was defined in the command media. Inputs, outputs, and resources were identified. Resources were categorized as critical, noise, standard operating procedure and controllable. The Process Map was extremely useful because it quickly highlighted duplicate activities, where implementation deviated from the documented process, and was used as an input to the Failure Mode and Effect Analysis (FMEA) and Design of Experiments (DOE).

The FMEA leveraged the process steps from the Process Map to identify potential failure modes with each process step, the effect of the failure, the cause of the failure, and any current detection mechanism. A numerical value was placed on each of these attributes and a cumulative Risk Priority Number (RPN) was assigned to each potential failure. The highest RPNs were the potential failures that needed to be mitigated or eliminated first and would eventually become the factors for the DOE.

The Product Scorecard contained all of the quantifiable data relating to the peer reviews. It showed the number of defects introduced and detected by phase, both in raw numbers, percentage, and by effort. Using Pareto Charts, it was easy to determine where defects entered the process, where defects were found by the process, and even which phases had the most impact (rework) to the bottom line. Surprisingly, 58% of the total detects were found in test, well after the product is deemed "done". Additionally, three phases accounted for greater than 92% of rework due to defects.

Phase Detected

Phase Introduced	Planning	Customer	Rqmts. Analysis	Design	Implementation	Test	Formal Test	Customer Before	TOTAL	Leaked
Planning	2.03	0	0	0	0	0	0	0	2.03	0
Customer	0	1.8	1.4	0	4.06	0	16.15	0	23.41	21.61
Rqmts. Analysis	0	0	7.32	12.04	32.77	41.6	56.09	0.79	150.61	143.29
Design	0	0	0.13	41.99	8.2	23.2	118.94	5.28	197.74	155.75
Implementation	0	0	0.17	0.5	154	90.3	88.88	23.3	357.15	203.15
Test	0	0	0	0.16	0.03	19.92	4.5	0	24.61	4.69
Formal Test	0	0	0	0	0	2.34	149.25	0	151.59	2.34
Customer Before	0	0	0	0	0	0	2.7	13.6	16.3	13.6
TOTAL	2.03	1.8	9.02	54.69	199.06	177.36	436.51	42.97	923.44	544.43

Figure 3

An improvement goal was set by the organization. The immediate goal was set around finding the defects earlier in the lifecycle rather than trying to reduce the number of defects. If the process could be improved to find the defects just one phase earlier in the lifecycle, the result would be many hundreds of thousands of dollars to the bottom line!

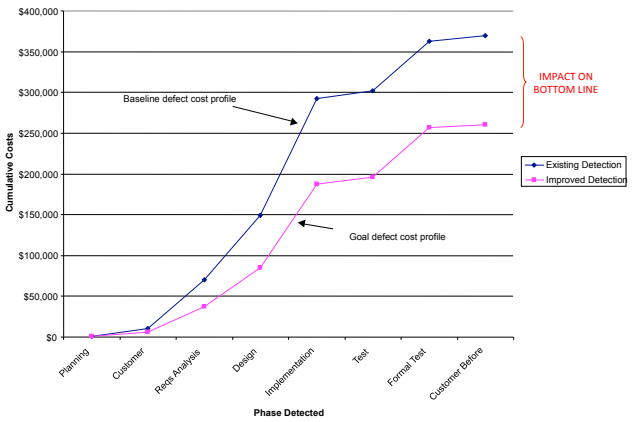


Figure 4

Same number of total defects introduced in the same phases

The data from the Product Scorecard was plotted to create a distributional characteristic of the process capability. Visually, this highlighted the lifecycle phases that were well below our goal of finding 80% of defects in phase, as seen in the figure below.

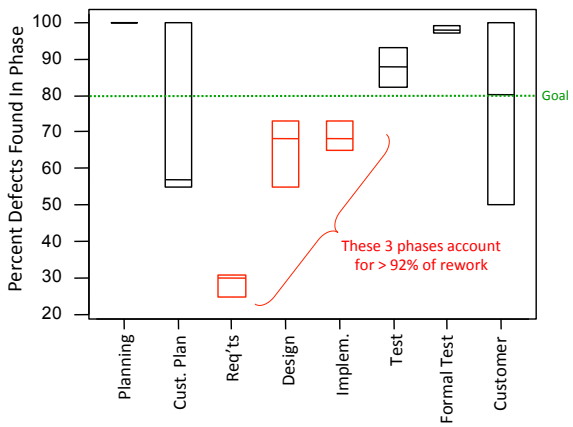
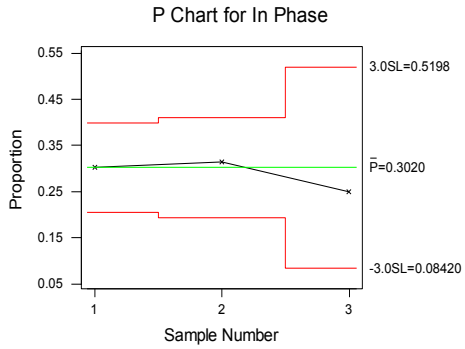
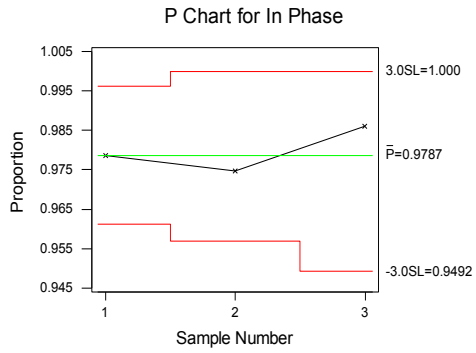


Figure 5

Going into this project, my belief was that a program could be identified that was conducting peer reviews effectively across the entire lifecycle and that program's process could be replicated across the organization. The Control Chart showed something quite different. All the programs were conducting peer reviews consistently, but the variation between lifecycle phases ranged widely. When the data was rationally sub-grouped by phase, the data became stable (predictable) within the subgroups, but there was extensive variation between the subgroups. This meant the variation came from the lifecycle phases not the programs. It would not be as simple as finding the program that conducted effective peer reviews and replicating its process across the organization.





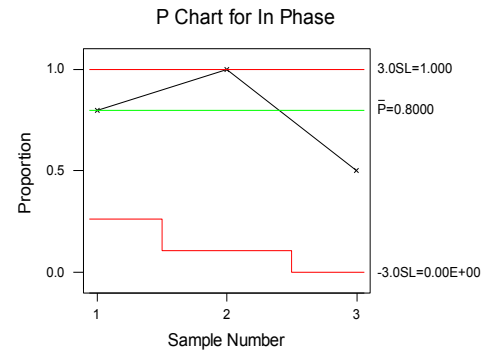


Figure 6

The Analysis of Variation confirmed that 72% of the process variation was between the subgroups (lifecycle phases) and only 28% was within the subgroup (programs). Since the data was only a sample of the population, Confidence Intervals were conducted to find out the true range of the population. This quickly showed that for the Requirements Phase, the best the process was capable of achieving was detecting 37% of defects in phase. In fact, if no action was taken it was 95% certain that the Requirements Phase will find between 21% - 37% of defects in phase, the Design Phase will find between 42% - 88% of defects in phase, the Implementation Phase will find 59% - 78% of defects in phase. This helped focus where to concentrate the improvement resources.

Analysis of Variance for percent

Source	DF	SS	MS	F	P
phase	7	10841.9583	1548.8512	8.701	0.000
project	16	2848.0000	178.0000		
Total	23	13689.9583			

Variance Components

Source	Var Comp.	% of Total	StDev
phase	456.950	71.97	21.376
project	178.000	28.03	13.342
Total	634.950		25.198

Project variation

Phase variation

Figure 7

Remember the high RPNs from the FMEA? These were used as the factors in a DOE. There were four factors (experience, training, review criteria, and number of reviewers). The response variable for the DOE was the percentage of defects found in a peer review. There were 16 runs, which made it a half-factorial DOE.

There were some limitations with this DOE. The products reviewed were different for each run; there were restrictions on randomization; and by the latter runs it was hard to find a peer review team that fulfilled the factor levels. For example, once somebody was trained they could not be untrained.

When analyzing data, always think golf (PGA = practical, graphical, and analytical). Practical analysis looked at the result of each for anything of interest. It was not until then the runs were sorted by response did any trends appear. The highest five runs all had no criteria, the lowest four consisted of inexperienced team members and six of the top seven were trained teams.

StdOrder	RunOrder	Program	Experience	Training	Criteria	Num People	%Match
6	6	-1	1	1	-1	1	80
10	14	1	1	1	-1	-1	80
14	13	1	-1	1	-1	1	80
1	1	-1	1	-1	-1	-1	70
2	5	-1	-1	1	-1	-1	70
16	16	1	1	1	1	1	70
4	7	-1	1	1	1	-1	65
7	4	-1	1	-1	1	1	60
8	8	-1	-1	1	1	1	60
5	2	-1	-1	-1	-1	1	55
13	10	1	1	-1	-1	1	55
11	12	1	1	-1	1	-1	55
12	15	1	-1	1	1	-1	55
9	9	1	-1	-1	-1	-1	45
15	11	1	-1	-1	1	1	35
3	3	-1	-1	-1	1	-1	30

Might have something here

Figure 8

Graphical analysis included a normal probability plot and a Pareto chart of the main effects, two-way and three-way effects. This clearly showed that training, criteria, and experience were the influential factors.

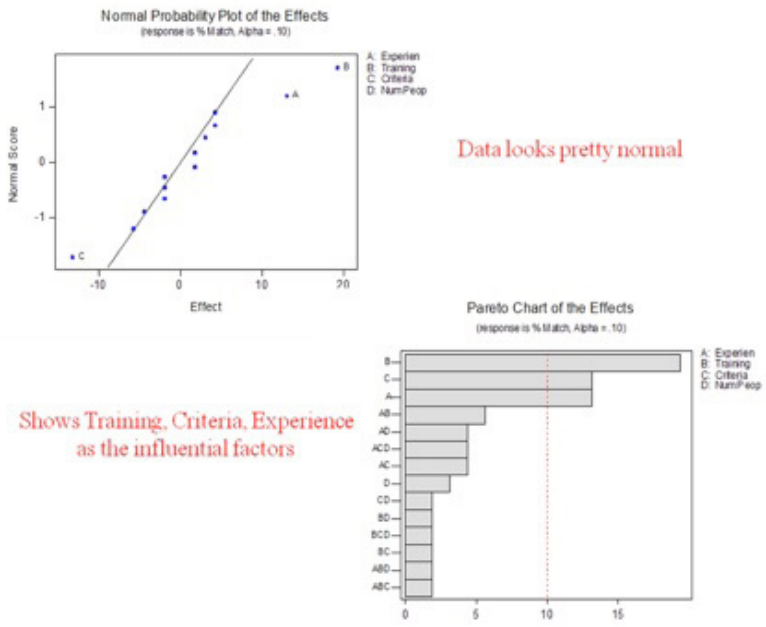


Figure 9

Analytical analysis not only showed the same influential factors but also quantified the effect and indicated whether to set the factor high or low. Training was the most influential, followed closely by experience. The process was relatively robust with respect to the language and number of people. If peer reviews are just as effective with half the people, this alone could have a big savings to the bottom line. The eye-opener here was that the peer review process was more effective without criteria. This went against intuition, but was based on data.

Fractional Factorial Fit

Estimated Effects and Coefficients for % (coded units)

Term	Effect	Coef
Constant		60.312
Program	-1.875	-0.938
Experien	13.125	6.562
Training	19.375	9.687
Criteria	-13.125	-6.562
Num Peop	3.125	1.562
Program*Experien	-1.875	-0.937
Program*Training	4.375	2.187
Program*Criteria	1.875	0.937
Program*Num Peop	-1.875	-0.937
Experien*Training	-5.625	-2.812
Experien*Criteria	4.375	2.188
Experien*Num Peop	-4.375	-2.187
Training*Criteria	-1.875	-0.938
Training*Num Peop	1.875	0.937
Criteria*Num Peop	1.875	0.937

Shows same thing Training, Criteria, Experience as the influential factors

Analysis of Variance for % (coded units)

Source	DF	Seq SS	Adj SS	Adj MS	F	P
Main Effects	5	2932.8	2932.8	586.56	*	*
2-Way Interactions	10	440.6	440.6	44.06	*	*
Residual Error	0	0.0	0.0	0.00		
Total	15	3373.4				

Figure 10

	Planning	Customer	Rqmts Analysis	Design	Implementation	Test	Formal Test	Customer Before	TOTAL
Planning	100%	0%	0%	0%	0%	0%	0%	0%	1%
Customer	0%	57%	10%	0%	10%	0%	24%	0%	1%
Rqmts Analysis	0%	0%	30%	7%	14%	13%	35%	0%	10%
Design	0%	0%	0%	68%	17%	6%	8%	0%	24%
Implementation	0%	0%	0%	2%	68%	13%	14%	3%	17%
Test	0%	0%	0%	1%	9%	88%	11%	0%	14%
Formal Test	0%	0%	0%	0%	0%	2%	98%	0%	31%
Customer Before	0%	0%	0%	0%	0%	0%	20%	80%	0%
TOTAL	1%	1%	3%	18%	17%	18%	40%	1%	100%

Percentage of defects identified by phase introduced/phase detected (from Product Scorecard)

AFTER IMPROVEMENTS

PHASE DETECTED

PHASE INTRODUCED	Planning	Customer	Rqmts Analysis	Design	Implementation	Test	Formal Test	Customer Before	TOTAL
Planning	98%	1%	0%	1%	0%	0%	0%	0%	21%
Customer	0%	40%	40%	13%	7%	0%	0%	0%	5%
Rqmts Analysis	0%	1%	88%	7%	4%	1%	0%	0%	16%
Design	0%	0%	1%	60%	38%	1%	0%	0%	25%
Implementation	0%	0%	0%	1%	94%	5%	0%	0%	31%
Test	0%	0%	0%	7%	7%	81%	4%	0%	2%
Formal Test	0%	0%	0%	0%	0%	0%	67%	33%	0%
Customer Before	0%	0%	0%	0%	0%	0%	0%	100%	0%
TOTAL	20%	2%	16%	17%	40%	4%	0%	0%	100%

Figure 11

	Planning	Customer	Rqmts. Analysis	Design	Implementation	Test	Formal Test	Customer Before	TOTAL	Leaked
Planning	2.03	0	0	0	0	0	0	0	2.03	0
Customer	0	1.8	1.4	0	4.06	0	16.15	0	23.41	21.61
Rqmts. Analysis	0	0	7.32	12.04	32.77	41.6	56.09	0.79	150.61	143.29
Design	0	0	0.13	41.99	8.2	23.2	118.94	5.28	197.74	155.75
Implementation	0	0	0.17	0.5	154	90.3	88.88	23.3	357.15	203.15
Test	0	0	0	0.16	0.03	19.92	4.5	0	24.61	4.69
Formal Test	0	0	0	0	0	2.34	149.25	0	151.59	2.34
Customer Before	0	0	0	0	0	0	2.7	13.6	16.3	13.6
TOTAL	2.03	1.8	9.02	54.69	199.06	177.36	436.51	42.97	923.44	544.43

Cost of Rework due to Defects in Days (from Product Scorecard)

AFTER IMPROVEMENTS

PHASE DETECTED

PHASE INTRODUCED	Planning	Customer	Rqmts Analysis	Design	Implementation	Test	Formal Test	Customer Before	TOTAL	Leaked
Planning	16.45	2	0	0.51	0	0	0	0	18.96	2.51
Customer	0	3.3	15.4	7	8.12	0	0	0	33.82	30.52
Rqmts Analysis	0	1	19.32	11.18	7.91	3.2	0	0	42.61	23.29
Design	0	0	0.39	22.49	11.1	1.6	0	0	35.58	13.09
Implementation	0	0	0	0.2	239.4	42	0	2.33	283.93	44.53
Test	0	0	0	0.16	0.06	1.76	0.15	0	2.13	0.37
Formal Test	0	0	0	0	0	0	0.5	0.58	1.08	0.58
Customer Before	0	0	0	0	0	0	0	10.2	10.2	10.2
TOTAL	16.45	6.3	35.11	41.54	266.59	48.56	0.65	13.11	428.31	125.09

Figure 12

Further investigation revealed that the use of criteria was restricting what the reviewers were looking for in the peer reviews. Training was developed to educate the reviewers on how to use the criteria. The criteria became a living document and as defects were found the checklist were updated.

The results showed remarkable improvements. The number of defects introduced before and after improvements were in the same order of magnitude (1947 vs. 1166) so that comparisons could be made between the "before" and "after" states. If you only look at the percentage of defects found in phase, as a lot of organizations do, the results can be misleading. It shows that five of eight phases actually found fewer defects in phase. Analyzing the data this way assumes all defects are created equal (it takes the same amount of effort to fix the defect) and does not take into effect the number of phases the defect leaked.

If the defects are transformed into the amount of rework, a completely different profile is observed. In those five phases that found a smaller percent of defects in phase the amount of rework decreased by 75%. Looking at the three phases that accounted for 92% of the rework, the improvements are dramatic. It can be confidently stated that two of the three phases will exceed the goal of finding 80% of defects in phase. The third phase only allowed 1% of the defects to make it to test, whereas before the improvements, 14% made it to test. This reduced the rework from 156 days to a mere 13. Remember, measure what are you trying to improve—is it number of defects or rework?

The bottom line savings exceeded the goal by more than 20%. There was a nominal increase in cost in the early stage but, as can be seen by the graph, the cost of rework leveled off after the implementation phase. This means almost no defects leaked into the testing phase or beyond. Imagine your organization having no defects leak beyond the implementation phase. It can be done!

Disclaimer:

CMMI® and CMM® are registered in the U.S. Patent and Trademark Office by Carnegie Mellon University.

ABOUT THE AUTHOR

Tom Lienhard is a Sr. Principal Engineer at Raytheon Missile System's Tucson facility and a Six Sigma BlackBelt. Tom has participated in more than 50 CMM® and CMMI® appraisals both in DoD and Commercial environments across North America and Europe and was a member of Raytheon's CMMI Expert Team. He has taught Six Sigma across the globe, and helped various organizations climb the CMM and CMMI maturity levels, including Raytheon Missile System's achievement of CMMI Level 5.

He has received the AlliedSignal Quest for Excellence Award, the Raytheon Technology Award and the Raytheon Excellence in Operations and Quality Award. Tom has a BS in computer science and has worked for Hughes, Raytheon, AlliedSignal, Honeywell and as a consultant for Managed Process Gains.

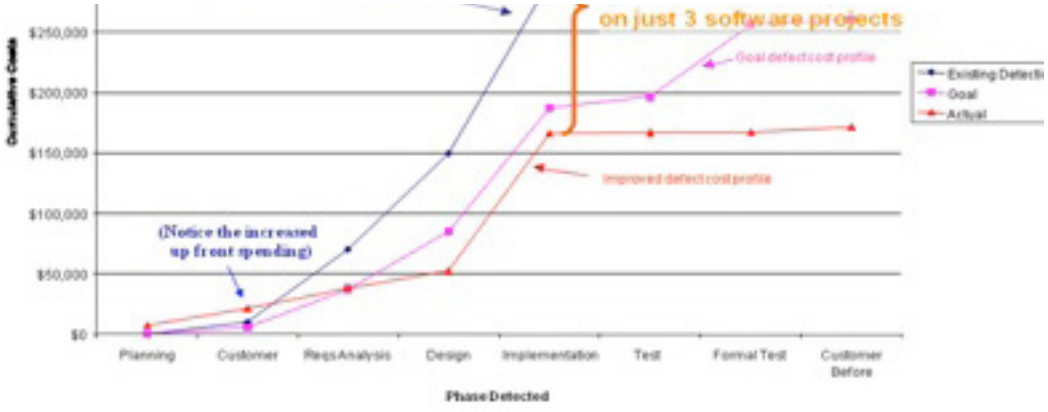


Figure 13