Improving Efficiency in Software Maintenance

Sergey Zeltyn*, Peri Tarr‡, Murray Cantor†, Robert Delmonico‡, Sateesh Kannegala§, Mila Keren*, Ashok Pon Kumar§, Segev Wasserkrug*

†IBM Rational Software
5 Technology Park Drive
Westford Technology Park
Westford, MA 01886 USA
mcantor@us.ibm.com

*IBM Haifa Research Lab
Haifa University
Mount Carmel, Haifa HA
31905 Israel
{keren,segevw,sergeyz}@il.ibm.com

‡IBM Watson Research
19 Skyline Drive
Hawthorne, NY 10532 USA
{rmd,tarr}@us.ibm.com

§IBM India Software Lab
Ground Floor, Manyata 'D2',
Nagawara, Outer Ring
Road, Bangalore, India
{ashokponkumar,
sateeshks}@in.ibm.com

ABSTRACT

Efficiency is critical to the profitability of software maintenance and support organizations. Managing such organizations effectively requires suitable measures of efficiency that are sensitive enough to detect significant changes, and accurate and timely in detecting them. *Mean time to close problem reports* is the most commonly used efficiency measure, but its suitability has not been evaluated carefully. We performed such an evaluation by mining and analyzing many years of support data on multiple IBM products. Our preliminary results suggest that the mean is less sensitive and accurate than another measure, percentiles, in cases that are particularly important in the maintenance and support domain. Using percentiles, we also identified statistical techniques to detect efficiency trends and evaluated their accuracy. Although preliminary, these results may have significant ramifications for effectively measuring and improving software maintenance and support processes.

Categories and Subject Descriptors

D.2.7 Distribution, Maintenance, and Enhancement, D.2.8 Metrics, D.2.9 Management, G.3 Probability and Statistics

General Terms

Algorithms, Management, Measurement, Experimentation

Keywords

Software maintenance, efficiency measure, defect handling time distribution, efficiency change detection, heavy-tailed distribution

1. INTRODUCTION

"No defects, no jobs. Absence of defects does not necessarily build business...something more is required." (W.E. Deming)

Defects are a fact of life for software and systems; it is impossible to ship defect-free software. Hence, s*ustaining engineering and maintenance* (SEM) business processes are put in place to fix defects and make small enhancements to address immediate end-user issues. The cost of SEM processes can be the difference between profit and loss for organizations; high after-delivery costs lead directly to losses on software sold. Controlling after-market expenses is, therefore, essential for SEM processes.

Thus, as Deming suggests, more than an absence of defects is required to make a profitable SEM business; it requires *efficiency*. The faster a SEM organization can close customer issues, the lower the after-market costs and the more profit it can earn. Not surprisingly, therefore, SEM organizations are under constant pressure to improve their efficiency.

Our work started with this challenge: how can we measure trends of efficiency in SEM processes and provide semi-automated support to help organizations identify significant changes in these trends quickly and accurately, diagnose causes of change, and evaluate the efficacy of changes adopted to improve efficiency? A key goal of our work is to identify *accurate*, *sensitive*, *timely*, and *actionable* measures of efficiency changes in SEM processes.

Towards this goal, we mined a database of customer problem reports filed by IBM customers over many years and across multiple products at different stages of their lifecycles. Our results were somewhat surprising: the handling times of these problem reports uniformly exhibit a heavy-tailed distribution, rather than a normal distribution. This led us to pose the following research question: Is the current common practice of using mean time to close defect reports an accurate and sensitive measure of changes in SEM process efficiency? Our results suggest the mean is less sensitive and accurate than another measure, potentially causing later detection of issues. This is particularly important for SEM processes, where feedback control loops must be short, as failure to identify and quickly address threats to efficiency can allow problems to spiral out of control. Our results may have significant ramifications for measuring and supporting SEM processes.

2. PROBLEM AND ANALYSIS OF DATA

To answer these questions, we started by performing an extensive analysis of defects reported by customers and fielded by SEM organizations within IBM. Our main data set covered defect reports for one of IBM's software divisions over several years. Overall, the data set contained tens of thousands of records across many products. We analyzed *defect handling times*, where handling time is the number of days that passed between opening and closing the report. We looked at both the handling time for all of this division's closed defect reports, for individual products and for specific defect severities. The defect handling time is well

MSR'11, May 21–22, 2011, Waikiki, Honolulu, HI, USA

accepted to be closely related to SEM efficiency, and mean handling time is currently a well-accepted measure of efficiency.

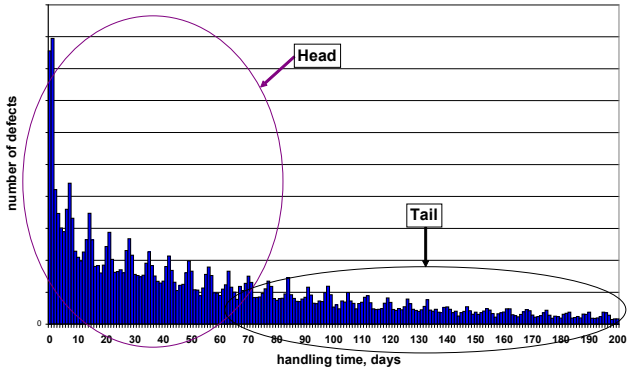

Figure 1: Histogram of defect handling times.

Our analyses all demonstrated the same phenomenon: that defect handling time shows a heavy-tailed distribution, rather than a normal distribution (Figure 1). Mathematically, this means that the tail of the corresponding density is not exponentially bounded. Informally, the heavy-tailed distribution occurs because some defect reports have very long handling times. Specifically, observations are not concentrated in the area of three standard deviations around the mean, as one often assumes for a normal distribution. For example, the mean handling time in the data set is 27 days, with a standard deviation of 59 days. However, about 0.5% of the defects took more than a year to close—almost six standard deviations from the mean—and others were open longer.[1] The longer handling times can reflect many factors, including changes deferred until the next product release.

We attempted to fit the well-known heavy-tailed distributions (e.g., Pareto, lognormal, Weibull) to our data, as this would have enabled us to use statistical methods tailored to the specific distribution. To do this, we performed a number of formal and graphical statistical tests (Kolmogorov-Smirnov test; P-P and Q-Q plots). We did not succeed: the handling time is not distributed, even approximately, according to any standard distribution.

> **Finding 1**: Handling times for SEM defect reports exhibit a heavy-tailed distribution. Moreover, this distribution is not one of the well-known heavy-tailed distributions.

Finding 1 implies that monitoring and trend detection of SEM process efficiency should be based on non-parametric techniques that do not use restrictive assumptions about handling-time distribution. Therefore, we sought to identify an appropriate efficiency measure and develop these non-parametric techniques.

2.1 Identifying Efficiency Measures for SEM

Because defect handling time is commonly accepted as closely related to SEM process efficiency, it makes sense to define efficiency measures based on this handling time. Figure 1 depicts the handling-time distribution for defects requiring a code change. We can distinguish two parts of this distribution: a "head," which contains the majority of defects, and a "tail," which contains defects that took significantly longer to fix.

We conducted an informal, preliminary set of discussions with IBM personnel who had experience with the management of SEM processes to understand (a) whether this distribution reflected their experiences, and (b) how the head and tail affect the assessment and management of their organizations' efficiency. A more careful study is left to future work, but the answers we heard were remarkably uniform. For question (a), not a single individual was surprised to see this distribution. Though they had not done the statistical analysis, they had learned to differentiate the efficiency of handling "normal" work from that of the outliers. For (b), the subjects told us that they do not ignore the tail, but they manage their organizations to maximize the efficiency of the head. That is, they sought to understand and optimize the handling time for "most" of the defects, and not allow the tail to have an undue effect on their processes. Some pointed out that the SEM organization cannot itself control the handling time for all defects in the tail. For example, some defects cannot be closed until a development team releases a new product version. A long handling time, therefore, need not reflect a SEM efficiency issue.

> **Finding 2**: Preliminary evidence suggests SEM processes are managed to optimize efficiency in the head, not the tail. Although the tail is relevant, problem reports falling in the tail need not reflect organizational efficiency and may, therefore, be treated differently from those in the head.

Finding 2 suggests that the tail should not have a substantial impact on accurate efficiency measures for SEM processes. This clear distinction between the head and tail of the handling-time distribution suggested to us, for several reasons, that it would be appropriate to evaluate *handling time percentiles* to measure SEM process efficiency, and compare its accuracy and sensitivity with handling time mean. First, percentiles explicitly separate head from tail and provide a measure of the width of the head (the upper bound on handling time). Indeed, percentile-based service-level agreements (SLAs) for waiting time or end-to-end time are widespread in many industries (e.g., [4]).

Second, we saw several cases where the heavy-tailed property of the distribution distorted the mean significantly, and in ways that affect the timely identification of efficiency issues. For example, Figure 2 summarizes quarterly handling-time data for defect reports that correspond to one product release. The high peak of the mean for the second quarter of 2007 is due to the closing of one old defect report. Suitable efficiency metrics for SEM should not be significantly influenced by such outliers. Percentiles generally are not prone to this problem, whereas the mean can be.

> **Finding 3**: Percentile of the handling time is much more stable than the mean with respect to very long handling times. These long handling times belong to the tail and should not affect efficiency measures.

Third, in contrast with the mean, the percentile metric is tunable: a user can choose the width of the head appropriate for a specific SEM process and consistent with organization SLAs. Based on our preliminary findings and on prior work, we use a default value of 80% at present, as SLAs based on handling 80% of issues are widespread (e.g., [4]). Further research is required to evaluate the 80 percentile as a default, and to determine how best to choose appropriate percentile values in different circumstances.

[1] For a normal distribution, the probability of exceeding six standard deviations from the mean is around 10^{-9}.

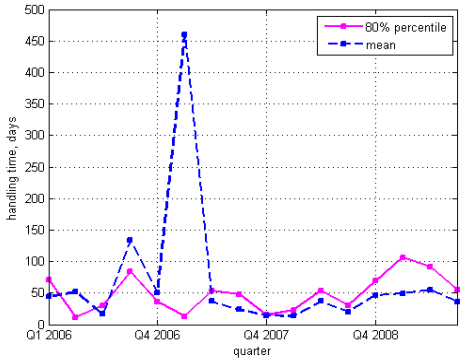

Figure 2: Defect handling time for one product release.

As noted earlier, a key characteristic of suitable measures of SEM efficiency is to be sensitive enough to detect efficiency changes accurately. Having thus identified percentiles as a measure that may have this characteristic, we performed a partial preliminary evaluation of the accuracy of using statistical tests based on handling time percentile vs. handling time mean. In particular, we looked at how such tests compare in a number of settings with respect to *false negatives* (the underlying handling time distribution changed, but the statistical test indicated that it did not), and *false positives* (the underlying handling time distribution did not change, but the statistical test indicated that it did).

Formally, we carried this out by formulating and evaluating two null hypotheses. Consider two handling-time samples that correspond to two time periods and constitute realizations of unknown underlying handling-time distributions.

H1: The handling-time mean of the underlying distribution did not change between two time periods.

H2: The handling-time percentile of the underlying distribution did not change between two time periods.

These lead to two important research questions:

R1: Can we establish *theoretical conditions* where the false negative rate for H2 is significantly smaller than for H1, given similar changes in mean and percentile of the two samples, and where the false positive rates are comparable?

R2: If R1 is correct, do handling-time distributions for actual SEM data satisfy these conditions as well?

Detailed analysis of R1 and R2 is deferred to another paper, but we present a preliminary partial analysis here. Note that R2 cannot be answered directly via real-data analysis. We do not know handling-time underlying distributions; thus, we cannot verify independently whether H1 and H2 are true for actual data. Hence, we used simulation experiments to explore these issues.

Because very large observations can significantly increase the sample variance and negatively affect the power of classical tests for means, we suggest that distributions satisfying R1 should be looked for in the heavy-tailed domain. Therefore, in our experiments, we used theoretical heavy-tailed distributions (Pareto and lognormal) and empirically-based heavy-tailed distributions that were simulated via bootstrapping actual SEM data. We ran two tests. The first test calculated false positive rates by generating two independent and identically distributed samples. In the second test we generated a third sample from a different distribution by scaling an original distribution by some factor to estimate the false negative rate. On this generated data, we evaluated H1 via standard two-sample t-test with unequal variances, while H2 was evaluated via a non-parametric test to detect a statistically significant change of the percentile [2].

In all cases, the theoretical confidence levels were equal to 95% and the simulated false positive rates for means and percentiles are within 5%. The false negative rates, however, showed some marked differences. Tables 1-3 show the false negative results. Each entry was computed using 10,000 simulation experiments.

Table 1: False negative rate for Pareto(2,3) and Pareto(4,3)

Statistical Test	Sample size		
	100	200	1,000
Means	54.47%	47.47%	39.09%
Percentile	48.61%	3.87%	0.00%

Table 2: False negative rate for Lognormal(1,1) and Lognormal(1,1+ln(2))

Statistical Test	Sample size		
	100	200	1,000
Means	78.78%	63.13%	18.26%
Percentile	81.08%	46.04%	0.05%

Table 3: False negative rate for bootstrapped SEM data

Statistical Test	Sample size		
	200	500	1,000
Means	46.98%	31.01%	10.37%
Percentile	64.53%	22.41%	1.83%

Clearly, percentiles consistently show fewer false negatives than means on larger samples for heavy-tailed distributions—sometimes strikingly so. We know that the false negative rate for any reasonable statistical test improves if the sample size increases. However, these results show that this improvement is very slow for the means test. A plausible explanation is that even for large sample sizes, outlier observations considerably increase the sample variance. In these cases, the t-test does not reject H1.

> **Finding 4**: For large samples from empirical and theoretical heavy-tailed distributions, percentile handling time can be more accurate than mean handling time with respect to change detection (false negatives).

Findings 3 and 4 provide a preliminary characterization of the circumstances under which percentile handling time is more accurate than mean time. Finding 3 is important mainly for small-to-moderate samples, where outliers strongly affect mean. Finding 4 is crucial for large samples where outliers affect variance. Additional research is required to completely characterize this.

As noted earlier, the feedback control loop for SEM organizations must be short to prevent efficiency issues from having serious consequences. Therefore, Finding 4 suggests that percentile handling time can be more appropriate for timely identification of efficiency issues that affect SEM organizations: higher false negative rates increase the time until change detection.

2.2 Statistically Significant…but is it *Right*?

Section 2.1 shows that percentiles can identify statistically significant efficiency changes in SEM data more accurately than the mean in some important cases. A key question is, do those statistically significant changes correspond to what SEM process participants or experts would identify as "real" efficiency issues?

To begin to evaluate this question, we created a tool, SEMinal, which analyzes SEM data for different products and helps users identify and diagnose efficiency issues. A detailed discussion of SEMinal is beyond the scope of this paper. However, to obtain some preliminary insight into the utility of the tool and the accuracy of percentiles in identifying user-visible efficiency issues, we conducted a small-scale pilot study in an IBM SEM organization. We selected five products at different lifecycle stages, imported their problem report data into SEMinal, and reviewed the results with the team leads. This study was informal and preliminary, but a few items emerged prominently. First, the data for all five projects clearly showed the heavy-tailed distribution described earlier. Second, in some cases, the 80 percentile measure was more stable than the mean, as suggested by Finding 3. We have not found projects where percentile was less stable than mean. Third, many of the statistically significant efficiency changes the tool identified were confirmed by the team leads as corresponding to efficiency-relevant events. We plan a more rigorous user study, but preliminary results are promising.

3. RELATED WORK

Considerable prior work exists in the field of software maintenance and evolution [1]. [6] describes the high cost of SEM that persists today. Those costs are the most significant driving factor for efficiency in SEM organizations today.

Prior work that describes a handling time distribution for customer defect reports is scarce (e.g.,[5]). [9] mentions "heavily-skewed long-tail" defect distribution. Note also that some prior work (e.g., [7]) identified the prevalence of power laws and various forms of heavy-tailed distributions, such as Pareto and log-normal, in many aspects of software/systems engineering.

In addition, there exists a significant body of work on prediction of defect handling time (e.g., [9],[11]) and on factors that affect various defect characteristics, such as size of code changes [8].

Mean handling time is a key performance measure in the software maintenance literature (e.g.,[12]). However, we are not familiar with any prior work that evaluated its suitability as SEM process efficiency measure or evaluated it against other measures.

"Recidivism rate," or the fraction of wrong fixes, was identified as important in evaluating efficiency [12] and clearly must be addressed in efficiency measures. We plan to do so in future.

Prior work on methodologies for change detection (e.g., [3]) advocates Statistical Process Control (SPC) for software engineering. We do not to follow this approach. A major problem with SPC, in our opinion, is that it is based on the existence of "process-in-control" that is used to derive process mean, standard deviation or other characteristics. This assumption is natural in industrial environments, but not in our SEM processes.

4. CONCLUSIONS

SEM process efficiency is critical to enterprises that deliver software. Managing the process effectively requires suitable measures of efficiency that are sensitive enough to detect significant changes, and accurate and timely in detecting them.

Our work contributes some preliminary results which, if validated rigorously, may have significant ramifications for effectively measuring, managing, and improving SEM organizations and their processes. We have mined and analyzed many years of real SEM data from multiple IBM products and provided what we believe is the first characterization of the handling times of that data as an unusual form of heavy-tailed distribution. This distribution, along with input from SEM experts indicating that efficiency of the head is more important than the tail, caused us to question the suitability of the commonly used mean handling time as an efficiency metric. Our early statistical analyses and user study strongly suggest that percentile handling time is more appropriate than mean handling time for the SEM domain.

5. ACKNOWLEDGMENTS

We gratefully acknowledge the contributions of Karthikeyan Dakshinamurthy, Atul Gohad, Kiran Mallekoppa, Sugam Mehta, Padmashree Mudhol, Balasubramani Radhakrishnan, Arthur Ryman, Sundari Sadasivam, Peter Santhanam, Kalpesh Sharma, Ramaprasad Srinivasachar, Mahesh Sundaram, Premkumar Swaminathan, Shaji Vaidyan, and Clay Williams.

6. REFERENCES

[1] Bennett K.H., Rajlich V.T. 2000 Software Maintenance and Evolution: A Roadmap. In *Proc. of the Conference on the Future of Software Engineering (ICSE 2000)*.

[2] Bristol D.R. 1990. Distribution-free confidence intervals for difference between quantiles. *Statistica Neerlandica*, 44 (2).

[3] Card D. 1994. Statistical Process Control for Software? *IEEE Software*, 11 (3), 95-97.

[4] Cleveland, B., Mayben J. 1997. Call Center Management on Fast Forward. Call Center Press.

[5] Kim S., James E. W., Jr. 2006 How long did it take to fix bugs? *MSR 06 proceedings.*

[6] Lientz B. P., Swanson E. B. Software Maintenance Management. Addison Wesley, Reading, MA, 1980.

[7] Louridas, P., Spinellis, D., and Vlachos, V. 2008. Power Laws in Software. *ACM Trans. on Software Engg.*, 18(1).

[8] Mockus A., Votta L. G. 2000. Identifying Reasons for Software Changes Using Historic Databases. *IEEE Intl. Conf. on Software Maintenance*, 120-130.

[9] Panjer L. D. 2007 Predicting Eclipse Bug Lifetimes, *MSR 07*

[10] Wasserkrug S. et. al. 2008. Creating Operational Shift Scheduling for Third Level IT Support: Challenges, Models and Case Study. *Int. J. Services Operations& Informatics* 3.

[11] Weiss C., Premraj R., Zimmermann T., *Zeller A. 2007. How Long Will It Take to Fix This Bug? MSR 07*

[12] E. Weller, 2000. Applying Quantitative Methods to Software Maintenance, ASQ Software Quality Professional, 3 (1).