A Quantitative Comparison of Corrective and Perfective Maintenance

Joel Henry and James Cain

Department of Computer and Information Sciences

East Tennessee State University

Summary: *This paper presents a quantitative comparison of corrective and perfective software maintenance activities. The comparison utilizes basic data collected throughout the maintenance process. The data collected are extensive and allow the impact of both types of maintenance to be quantitatively evaluated and compared. Basic statistical techniques test relationships between and among process and product data. The results show interesting similarities and important differences in both process and product characteristics.*

1. INTRODUCTION

Most large software systems have long lifetimes during which the software undergoes significant change. Software maintenance is defined as the set of activities performed to change a software product after the software product is delivered to the customer (Pressman, 1987). These activities, plus the tools and methods used to maintain software are referred to as the maintenance process. Changes to existing software include adding functionality to the software, correcting defects discovered in the software system, adapting the software to changes in the environment, and changing the software to support future maintenance or operation. The variety of changes made to software and the fact that most maintenance personnel were not involved in the development effort add significantly to the difficulties encountered while performing software maintenance.

In recent years the software process (including both development and maintenance) has received a great deal of attention (Humphrey et al., 1987) (Humphrey, 1989) (Bollinger et al., 1991) because the process used to develop and maintain software significantly impacts the cost, quality and timeliness of software products. The impact is so significant that software process improvement is seen as the most important approach to software product improvement (Humphrey, 1989).

PRECEDING PAGE BLANK NOT FILMED

While software development typically refers to the creation of new software, software maintenance is performed for a variety of reasons. The four types of software maintenance activities are:

1. Corrective - changes made to correct defects in software
2. Adaptive - changes needed to adapt existing software to a changing environment
3. Perfective - enhancements to software which provide additional functionality or modify existing functionality
4. Preventative - changes which improve future maintainability, reliability or support future enhancements

The tasks employed during maintenance are very similar to those applied during development: specify, design, code, and test. Thus, the first step in maintenance is to obtain a written specification of the functionality to be added. The written specification is given by changes and additions to the documentation specifying the functionality of the existing software. In principle the written specification is given completely and is never changed during the ensuing maintenance effort. In practice, however, these specifications are corrected and refined throughout the maintenance process. The changing of functional specifications during maintenance and development is referred to as requirements volatility. Requirements volatility has been cited as the leading problem in a field study of software managers (Thayer et al., 1982). Changing requirements adversely affects the design, coding and testing of software. An acute need exists to quantitatively assess the maintenance process and the impact of requirements volatility on both the maintenance process and the software product.

The focus of this paper is a comparison of corrective and perfective maintenance activities driven by changes to the specification documents of existing software. This comparison attempts to answer three general questions:

1. What similarities exist between corrective and perfective maintenance characteristics?
2. What differences exist between corrective and perfective maintenance characteristics?
3. What do these similarities and differences suggest about the nature of perfective and corrective maintenance?

This paper describes a portion of the results of a three-year study conducted at a large commercial software organization to assess the maintenance process and the impact of requirements volatility on the maintenance process. The portion of the assessment described here illustrates similarities and differences between corrective maintenance and perfective maintenance.

While this paper describes the results obtained within a single large organization, the results may be used by other organizations. These results indicate organizations should manage corrective and perfective maintenance differently.

The remainder of this paper is divided into two sections. Section 2 presents analysis results in five distinct areas. Section 3 outlines conclusions and the direction of future work.

2. ASSESSMENT RESULTS

Five significant results are described in the following subsections. Each subsection discusses the focus of the analysis, the data used in the analysis, and the statistical results. A maximum P-value of 0.05 and the minimum R^2 value of 0.75 were established as criteria for asserting relationships existed. This maximum P-value represents a 5% chance of mistakenly assuming a relationship exists. The minimum R^2 can be viewed as explaining 75%of the variability of the predicted variable.

2.1 CORRECTIVE AND PERFECTIVE SIMILARITIES

2.1.1 PRODUCTIVITY

Software maintenance productivity is of particular interest when examining corrective and perfective maintenance activities. We compared the productivity of both types of activities using corrective and perfective activity measures. Productivity is measured in SLOCs (source lines of code) per day and changed SLOCs per day.

Our initial examination showed only a 5.6% difference in productivity, with perfective maintenance being slightly more productive. Requirements volatility, tracked by specification changes occurring during design, code, and test, showed only an 8.5% difference. Again, perfective maintenance productivity was slightly higher.

The Mann-Whitney test, which statistically tests the differences in the sample means, was applied in order to test the hypothesis that corrective and perfective maintenance items are similar. The Mann-

Whitney test produced a P-value of 0.9833 which is not less than the previously established maximum P-value of 0.05. The P-value of 0.9833 supports acceptance of the hypothesis that the productivities of corrective items and perfective items are not statistically different.

2.1.2 SIGNIFICANT IMPACT ON PRODUCTIVITY

The previous section strongly supports the assertion that productivity of corrective maintenance and perfective maintenance is not statistically different. However, we noted differences between corrective and perfective product impact, as shown in Table 1. Perfective maintenance impact is greater in terms of SLOCs changed and modules changed than corrective maintenance. SLOCs changed per module appear similar. We investigated which of these three factors influenced productivity the most. We found the most significant factor influencing productiviy is SLOCs per module.

	CORRECTIVE TOTAL SLOCS	CORRECTIVE MODULES CHGD	PERFECTIVE TOTAL SLOCS	PERFECTIVE MODULES CHGD
MEAN	33.1905	1.7541	150.8511	3.0459
STD DEV	55.3804	1.7763	517.6439	3.6676
MEDIAN	10.5000	1.0000	23.5000	2.0000

Table 1. Basic Statistics for Corrective and Perfective Characteristics

	CORRECTIVE PRODUCTIVITY	PERFECTIVE PRODUCTIVITY
SLOCS per MODULE	0.951	0.788

Table 2. Linear Correlations of Product Impact vs Productivity

Table 2 gives the linear correlations for productivity with SLOCs changed per module for both corrective and perfective maintenance. The linear correlations for corrective and perfective are both above the 0.75 threshold. These correlations suggest corrective and perfective maintenance productivity are significantly influenced by the distribution of change across modules.

4.2 CORRECTIVE AND PERFECTIVE DIFFERENCES

4.2.1 PRODUCT IMPACT

This section describes the significant differences between corrective and perfective maintenance. The characteristics compared include size of the change (measured in SLOCS), implementation effort(measured in person days), and distribution of change(measured in modules changed). We again applied the Mann-Whitney test, testing the hypothesis that the size and distribution of change are similar for both types of maintenance.

The results of the Mann-Whitney tests for modules changed and size of change produced P-values of 0.0170 and 0.0012, both significantly less than the maximum P-value of 0.05. These P-values support rejection of the hypotheses that modules changed and size for corrective maintenance are similar to corresponding measures for perfective maintenance. Thus, there are more lines of code, and are more modules changed for perfective maintenance than for corrective maintenance.

4.2.2 PRODUCT IMPACT ON QUALITY

Thus far, analysis has focused on corrective and perfective characteristics within the maintenance process, prior to delivery to the customer. This subsection examines the product impact of corrective and perfective maintenance activities on software quality.

We obtained defect data gathered prior to delivery and following product delivery. These defects have different levels of severity and are of great importance to the customer. Defect data (pre-delivery and post-delivery) and product impact data were analyzed using rank correlations to determine, statistically, their relationships.

	CORRECTIVE CHANGED SLOCs	PERFECTIVE CHANGED SLOCs
PRE-DELIVERY DEFECTS	0.3214	0.9702
POST-DELIVERY DEFECTS	0.2143	0.8884

Table 3. Rank Correlations of Defects And Changed SLOCs

Table 3 presents the rank correlations between the corrective and perfective changed SLOCs and the number of pre-delivery and post-delivery defects detected. The number of perfective changed SLOCs has a much stronger positive correlation to both types of defects than the number of corrective changed SLOCs. These results suggest that as the number of perfective changed SLOCs increases, the number of pre-delivery and post-delivery defects also increases.

4.2.3 PROCESS IMPACT ON QUALITY

This subsection investigates the impact of productivity on the number of pre-delivery and post-delivery defects. This is an important area because the customer is not only interested in software maintenance being performed in a cost-effective, timely fashion, but also in the quality of the delivered software. In order to investigate the relationship between corrective and perfective productivity, rank correlations will again be used.

	CORRECTIVE PRODUCTIVITY	PERFECTIVE PRODUCTIVITY
PRE-DELIVERY DEFECTS	- 0.8214	0.4545
POST-DELIVERY DEFECTS	- 0.8214	0.5775

Table 4. Rank Correlations of Defects and Productivity

Table 4 presents rank correlations between productivity and quality for corrective and perfective maintenance. Perfective productivity has weak correlation with the number of pre-delivery and post-delivery defects detected, while corrective productivity has a very strong negative correlation with the number of pre-delivery and post-delivery defects detected. This implies that as corrective maintenance productivity increases, the number of defects increases.

3. CONCLUSIONS

The results of this investigation suggest several interesting, and perhaps provocative, characteristics of software maintenance. Viewing the similarities, differences, and statistical relationships between perfective and corrective maintenance confirms a previously advanced "rule of thumb", questions another such rule, and leads to the proposal of a new rule.

Requirements volatility analysis led to the discovery of some important differences between perfective and corrective. The size of change and distribution of change to the product differed significantly between perfective and corrective maintenance; perfective maintenance resulted in larger and more distributed change to the software product than corrective maintenance. However, productivity did not show a significant statistical difference because the average change per software module remained roughly the same for both types of maintenance. These results confirm the old rule: the more local the change to the software product, the easier the maintenance effort.

Analysis of the impact of perfective and corrective maintenance on the quality of the delivered software product provides two interesting results. First, strong positive rank correlation exists between the impact of perfective maintenance and the number of post-delivery defects detected in the software. This correlation suggests that as the impact of perfective maintenance increases the number of post-delivery defects also increases. Second, a strong negative correlation exists between the impact of corrective maintenance productivity and the number of pre-delivery and post-delivery defects. This correlation suggests that as the impact of corrective maintenance increases the number of post-delivery defects decreases. This result questions an old rule: fixing errors inserts new errors into software.

Our results suggest a new rule: as the impact of changes to the software product caused by corrections to the requirements document increase, the number of pre-delivery and post-delivery defects decreases. Obviously a realistic limit to this rule exists. The number of pre-delivery and post-delivery defects could not be eliminated by maximizing the impact of corrective maintenance.

These results illustrate two additional points. First, neither the size of the change nor the distribution of the change, taken individually, influence productivity. It is the combination of these factors which significantly impact the productivity of both perfective and corrective maintenance activities. Second, perfective and corrective maintenance differ significantly in both the impact on the software product and the impact on the number of defects. These two types of maintenance differ to the extent that they should be managed and assessed separately.

REFERENCES

Bollinger, T.B. and McGowen, C., "A Critical Look at Software Capability Evaluations," *IEEE Software*, July 1991.

Humphrey, W.S. and Sweet, W.L. "A Method for Assessing the Software Engineering Capability of Contractors," Software Engineering Institute, Carnegie Mellon University, September 1987.

Humphrey, W.S., *Managing the Software Process*, Addison-Wesley, 1989.

Jablonski, J., R., *Implementing Total Quality Management: An Overview*, Pfeiffer, 1991.

Pressman, R. S., *Software Engineering: A Practitioners Approach*, McGraw-Hill, 1987.

Thayer, R. H., Pyster, P. and Wood, R. C., "Validating Solutions to Major Problems in Software Engineering Project Management," *IEEE Computer*, August 1982.

A Quantitative Comparison of Corrective and Perfective Maintenance

Software Engineering Workshop
December 1, 1994

Joel Henry
Jim Cain

East Tennessee State University
Department of Computer and Information Sciences

Overview

- Introduction
- Process
- Data collection
- Quantitative comparison
    - Similarities
    - Differences
- Conclusions

Introduction

- Focus

  - assessment of corrective and perfective maintenance activities driven by changes to the specification documents

- Purpose

  - quantitative comparison of maintenance process and product impact

Process Terminology

- Items

  - Upgrade
  - Corrective

- Specification Changes (SCs)

  - Upgrade
  - Corrective

- Miscellaneous terms

  - SLOCs
  - Modules

Data Collection

- WHAT :
  - Process and product data
  - Corrective and perfective maintenance data

- HOW:
  - Item, specification change, and computer program change numbers

  - Validation performed by multiple groups

- WHERE:
  - Storage in a single, central, tightly controlled database

SIMILARITIES: PRODUCTIVITY

- Corrective Items vs. Perfective Items

  - Basic statistics showed only a 5.6% difference in SLOCS per person day

- Corrective SCs vs. Perfective SCs

  - Basic statistics showed only a 8.5% difference in SLOCS per person day

- Mann-Whitney Test showed no statistical difference in productivities

SIMILARITIES: SIGNIFICANT FACTOR

	CORRECTIVE ITEM SLOCS per PERSON DAY	PERFECTIVE ITEM SLOCS per PERSON DAY
SLOCS per Module	0.951	0.788

- Coorelations of corrective items and perfective items with SLOCs per module

DIFFERENCES: SIGNIFICANT FACTOR

	CORRECTIVE CHANGED SLOCs	PERFECTIVE CHANGED SLOCs
PRE-DELIVERY DEFECTS	0.3214	0.9702
POST-DELIVERY DEFECTS	0.2143	0.8884

- Corrective changed SLOCs show weak coorelation to pre-delivery and post-delivery defects

- Perfective changed SLOCs show significant coorelation to pre-delivery and post-delivery defects

DIFFERENCES: PRODUCTIVITY/DEFECT RELATIONSHIP

	CORRECTIVE PRODUCTIVITY	PERFECTIVE PRODUCTIVITY
PRE-DELIVERY DEFECTS	- 0.8214	0.4545
POST-DELIVERY DEFECTS	- 0.8214	0.5775

- Productivity of perfective maintenance shows weak coorelation with both pre-delivery defects and post-delivery defects
- Productivity of corrective maintenance shows a negative coorelation with both pre-delivery and post-delivery defects

Conclusions

- Productivity similar
- Change per module similar
- Process and product impact on quality differ