Food for Thought™

An e-newsletter published by
Software Quality Consulting, Inc.

April 2009 , Vol. 6 No. 3
[Text-only Version]

Welcome to ***Food for Thought™***, an e-newsletter from **Software Quality Consulting**. I've created free subscriptions for my valued business contacts. If you find this newsletter informative, I encourage you to continue reading. Feel free to pass this newsletter along to colleagues by clicking this **Forward Email** link. If you've received this newsletter from a colleague and would like to subscribe, please click this **Enter New Subscription** link. If you don't wish to receive this newsletter, click the **SafeUnSubscribe**™ link at the bottom of this newsletter, and you won't be bothered again.

Your continued feedback on this newsletter is most welcome. Please send your comments and suggestions to **info@swqual.com**.

In this issue

In **This Months' Topic**, I discuss the importance of delivering a quality product...

Regular features to look for each month are:

- **Monthly Morsels**
  Hints, tips, techniques and reference info related to this month's topic
- **Calendar**
  Conferences, workshops, and meetings of interest to software engineers, QA engineers and anyone interested in software development

This month's topic

Does Anyone Care About Quality?

Most consumers care about quality. When consumers buy products and services, there is an expectation of quality. When you go to the pharmacy for a cold remedy, you expect the product you buy will help you feel better, not make you sicker. When you buy a new car, you expect it will work as advertised, not stall on the highway. If you need technical support for say, a new laptop, you expect the support person will actually help you solve your problem, not give you a run-around.

For some reason, our expectations for software are much lower than for any other product. When we buy software, we expect it will be buggy. We seem to be resigned to the fact that software will have problems and there is nothing we can do about it.

> “...there’s ultimately one reason software companies keep selling buggy software. *Because we let them*.” [1]

The issue I would like to discuss this month is - does anyone care about quality?

Does your company have a reputation for quality?

Some companies are known for delivering quality products. Companies like H-P and Motorola are two examples of large companies known for producing consistently high quality software products. By working hard at integrating a quality culture into all aspects of software product development, these companies have earned the respect and the loyalty of many consumers. These companies have made a considerable investment in quality. The return on this investment is fiercely loyal customers. What could be better than that?

However, once a company’s quality reputation becomes tarnished, customers will choose someone else and it is often extremely difficult to win those customers back. You need only look at what’s left of the US auto industry to see firsthand evidence of this. A recent article in the Wall Street Journal stated that:

> “A perception of inferior quality is the most serious problem facing GM, aside from its financial predicament” [2]

Even though independent organizations such as Consumer Reports have given GM cars such as the Chevy Malibu high marks for quality, the perception is that many American made cars don’t compare to European and Asian cars with regard to quality and reliability. Having owned a Malibu back when it suffered from poor quality, I swore I would never buy another one.

Japanese automakers showed the world it was possible to build high quality cars for a reasonable price. The CEO of Ford, a huge fan of Toyota’s legendary quality practices, is trying to do for Ford what Toyota has done for the rest of the auto industry - instill a quality culture. Like GM, Ford’s reputation has been tarnished...

> In the 1970’s, Ford chose to ignore a lethal defect in the Pinto. This was as an example of corporate arrogance and disregard for consumers. This decision demonstrated that Ford did not care about quality or safety.

- Read more about the **Ford Pinto’s safety problems...**

Now how does all this relate to software? Several people have observed that there are striking similarities between the arrogance that typified the US automakers for several decades and the software industry. Think about it - what other industry routinely produces defective products, is aware of many of these defects, chooses to not fix them, and then has the audacity to charge customers for fixing these defects!

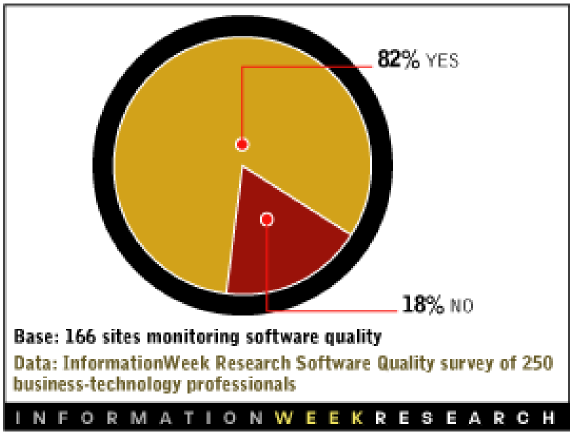
How did we get here? Software marketing people have long believed that quality does not sell and that only new features sell. If that were true, then:

- Why is **Google Search** - which ostensibly has only one feature - the most popular search engine in the world?

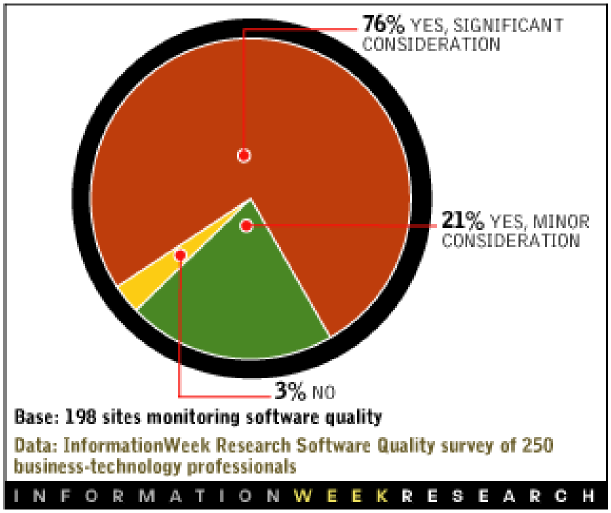
- Why have so many PC users refused to migrate to Vista? There's lots of new features in Vista but many of these people have finally realized that **we don't need all those new features - we need stability and quality!**

A recent article in *InformationWeek* magazine reported that:

> "Of the sites monitoring software quality in *InformationWeek* Research's Software Quality survey, 82% say they have avoided using a vendor because of a reputation for poor-quality code." [5]



> "More than three-quarters of the 198 companies in *InformationWeek's* survey monitoring the performance of their software apps say that software quality is a significant consideration when choosing a software vendor, and just one in five considers it a minor matter. Only 3% report that software quality isn't a factor when selecting a software vendor." [5]



In the current economy, I believe software companies who deliver lower quality products and services - whether intentionally or unintentionally - will be in worse shape when the recession ends than those who do deliver better quality products and services.

Why Care About Software Quality?

We need to care about software quality because software is finding its way into more and more aspects of routine daily life...

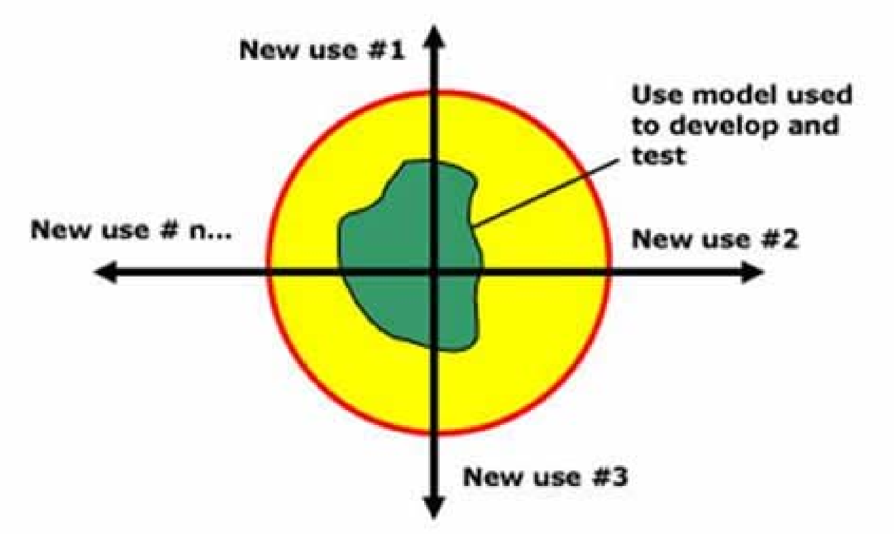
> “Software is pervasive in modern society, but we are often unaware of its presence until problems arise. Software is one ofthe most important and yet one of the most economically challenging technologies of the current era. As a purely intellectual product, it is among the most labor intensive, complex, and error-prone technologies in human history. Even though many successful software products and systems exist in the world today, an overall lack of attention to quality has also led to many problematic systems that do not work right as well as to many software projects that are late, over budget, or canceled.” [3]

A recent study by the US National Institute of Science and Technology (NIST) [6] estimated that software defects cost the US economy almost $60 billion a year. Given the increasing trends in application complexity, this number is certainly likely to increase...

So why does it seem that customers don’t care about software quality? Well, as Watts Humphrey observed:

“Because defective software works.” [7]

It may not work in every situation or for every user, but it usually works as long as users stay within defined use models. These defined use models represent the **best guess** of developers and testers as to how users will use the software. Watts Humphrey calls these defined use models the “testing footprint” [7] as shown below.

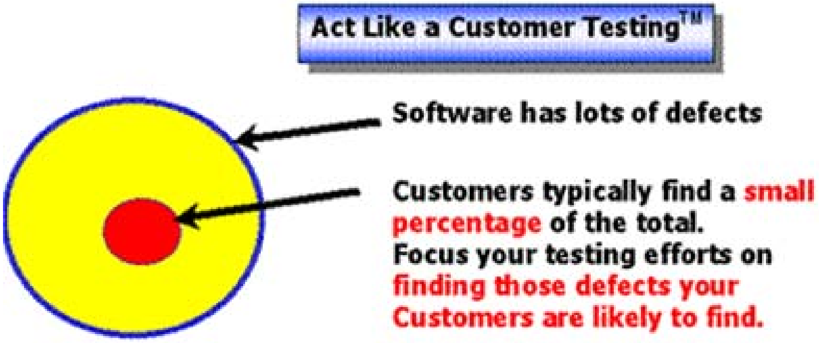


How can we develop high quality software when we know that zero-defects is not possible?

Given the exponential increase in complexity, everyone should understand that releasing software with zero-defects is not practical and not possible.

- To understand why, read more about **defective software...**

So what can we do? Well, understanding your customer’s testing footprint is very similar to a testing strategy I call - **Act Like a Customer Testing.** TM Test teams who use **Act Like a Customer Testing** TM often find that there are far fewer defects reported by customers. That doesn’t mean there aren’t defects in the software, but rather, your customers are not finding them.


Act Like a Customer Testing TM
Software has lots of defects
Customers typically find a small percentage of the total.
Focus your testing efforts on finding those defects your Customers are likely to find.

To incorporate this strategy into your testing, testers need **domain knowledge**. Armed with domain knowledge, testers can put themselves in their customer’s shoes and do what customers do. This helps find the most important defects - the ones your customers would find...

- Read more about **Act Like A Customer Testing™...**

A Measure of Quality

How can you tell how good a job you are doing at Acting Like a Customer in your testing? The most straightforward way is a simple metric developed by Capers Jones, called **Defect Removal Efficiency**. [4]

- Read more about **defect-removal efficiency metric...**

Simply count the number of defects found during your normal development and testing process and then divide that by that same number PLUS those defects reported by customers during some predefined usage period...

$$\frac{\textbf{Total defects we find}}{\textbf{Total we find + Customer-reported defects}}$$

Most shrink-wrap software has a Defect Removal Efficiency of between 70-80%. Best-in-class companies such as H-P and Motorola typically have defect removal efficiencies of 99.5% or higher. The higher the defect removal efficiency metric, the better your test cases are at finding those defects that customers are likely to find.

The Bottom Line....

Help your organization adopt a quality culture by developing a vocabulary for talking about quality in the context of your products and from the perspective of your customers. Once you have this vocabulary, derive a few simple measures - Defect Removal Efficiency is a good example. From these few measures, management can then begin to make better business decisions and your customers will thank you with their loyalty...

'Til next time...

Monthly Morsels

Every month in this space, you'll find additional information related to this month's topic.

References

1. Minasi, M., *Software Conspiracy: Why Companies Put Out Faulty Software, How They Can Hurt You and What You Can Do About It,* McGraw-Hill, 1999.

2. Helliker, K., and Stoll, J., "Malibu Shows Road to Recovery Is Bumpy", *Wall Street Journal*, April 1, 2009.

3. Krasner, H. “Using the Cost of Quality Approach for Software”, *CrossTalk*, November, 1998.

4. Jones, C., “Software Defect Removal Efficiency”, *IEEE Computer*, April 1996.

5. Weier, M. H. , “Consortium Confronts Software-Quality Issues”, ***InformationWeek***, May 26, 2003.

6. “The Economic Impacts of Inadequate Infrastructure for Software Testing”, *NIST Planning Report* 02-3, May 2002.

7. Humphrey, W., “**Defective Software Works**”, *news@sei newsletter*, Number 1, 2004.

Calendar

Every month you’ll find news here about local and national events that are of interest to the software community…

- **Software Quality Calendar**

  There are many organizations that sponsor monthly meetings, workshops, and conferences of interest to software professionals. **Find out what’s happening…**

- **Workshops Offered by Software Quality Consulting**

  Software Quality Consulting offers workshops in many topics related to software process improvement. **Get more info…**

About SQC

Software Quality Consulting provides consulting, training, and auditing services tailored to meet the specific needs of clients. We help clients fine-tune their software development processes and improve the quality of their software products. The overall goal is to help clients achieve Predictable Software Development™ – so that organizations can consistently deliver quality software with promised features in the promised timeframe.

To learn more about how we can help your organization, **visit our web site** or **send us an email**.

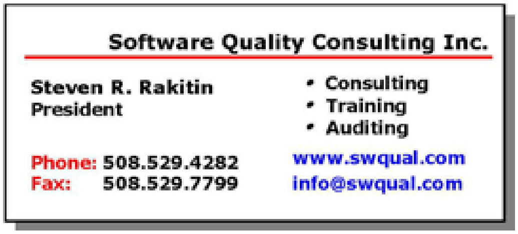
I hope this newsletter has been informative and helpful. Your comments and feedback are most welcome. **Send me your feedback…**

Thanks,

Steve

Steve Rakitin

info@swqual.com



Software Quality Consulting Inc.

Steven R. Rakitin
President

- Consulting
- Training
- Auditing

Phone: 508.529.4282
Fax: 508.529.7799

www.swqual.com
info@swqual.com

Graphic design by **Sarah Cole Design**.