White Paper
April 2009

CAST
ACHIEVE INSIGHT. DELIVER EXCELLENCE.

The Business Value of Application Internal Quality

An Executive's Guide to Immediate Business Impact

When a mission-critical application fails, the loss of business revenue is large and swift. Poor application quality causes highly-visible major outages; it also causes ongoing lapses in business performance that are less visible, but steadily add up to substantial revenue loss. Even minor quality improvements can result in significant business gain. Yet, executives struggle to build a business case to justify proactive investments in application quality. This paper presents a quantitative framework for measuring the immediate and positive revenue impact of improving application quality.

Dr. Bill Curtis
Senior Vice President and Chief Scientist, CAST

Co-author CMM, People CMM, & Business Process Maturity Model, Fellow of the Institute of Electrical and Electronics Engineers, authority on IT Software Quality

Executive Summary

Dr. Bill Curtis, SVP and Chief Scientist at CAST, presents a powerful quantitative framework for estimating and measuring how improvements in application quality boost business benefits and reduce IT costs.

The Problem

- *While most businesses can quantify the costs of their application failures, they struggle to build business cases to justify proactive investments in application quality to prevent these failures.*
- *Rarely are production-environment failures due to botched functional requirements – all to often they are caused by non-functional defects – defects that result from poor design and coding that lie hidden until they are ruthlessly exposed during business operations.*
- *Poor application quality creates several dangerous problems that put current and future business revenue at risk – outages in critical business systems, corrupted data, security breaches, and regulatory non-compliance. These problems usually escape detection during testing.*
- *The result: Death by a Thousand Cuts. Business performance problems due to poor application internal quality substantially decrease business value and increase IT costs throughout an application's useful life.*

The Solution

- *Through extensive research and real-world experience in serving 650+ enterprises worldwide, CAST has identified five key areas of internal software quality that most affect business outcomes – the CAST Application Health Factors.*
- *Improving these Application Health Factors has a direct impact on increasing business productivity, accelerating speed to market, improving customer experience, and decreasing IT costs. It's a win-win situation: substantial business benefit at lower IT cost.*

Executive Summary (continued)

- *By measuring and monitoring the Application Health Factors right from the start of an application's lifecycle, you ensure optimal business benefit delivery. Application quality problems are found well before they become major issues at deployment; the root causes of these problems are highlighted, and can be fixed once and for all, thus preventing significant business disruption and customer frustration.*
- *A 10% decline in application performance can quickly add up to almost half a million dollars of business productivity lost in a quarter.*
- *When internal quality improvements reduce rework by 25% a year and enable maintenance staff to spend 60% less time making sense of the code, it frees up $75,000 of effort per application for creating new business functionality.*

Key Takeaways

- *This paper provides concrete calculations of the itemized business and IT costs incurred due to a decline in application quality.*
- *By proactively measuring and improving Application Health Factors, IT teams can dramatically increase application quality and the business value delivered by an application while cutting IT costs.*
- *When losses due to poor software quality are computed only against development or maintenance costs, the full impact of the loss remains invisible to the business. However, when computed against business costs and lost business opportunities, a compelling business case can be made for investing in application quality.*
- *IT organizations can readily use the detailed examples contained in this white paper to create a compelling business case for improving the internal quality of mission-critical applications.*

Contents

I. Why Is the Internal Quality of a Business Application Important?

As companies conduct more of their critical business activities online, concern over the quality of IT applications has moved from the IT department to the boardroom. Downtime on a retail e-business Website can be measured in lost sales. Compromised customer records can be measured in customer defections. Corrupted financial data can be measured in restatement costs. All three can be measured in lines of bad press.

The types of application problems that most frequently cause these disasters are not defects in functional logic – 'what the application is supposed to do'. Rather they are caused by the non-functional defects that result from poor design and coding – '*how* the application does it'. Most functional defects are detected during development through reviews and testing. The non-functional defects often lie hidden beneath the surface of an application until they are triggered during ordinary business operations and cause embarrassing calamities. As Diomidis Spinellis points out in his recent book, *Code Quality*, "a failure to satisfy a non-functional requirement can be critical, even catastrophic…non-functional requirements are sometimes difficult to verify. We cannot write a test case to verify a system's reliability."

While most businesses can quantify the costs of their application failures, they struggle to build business cases necessary to justify proactive investments in application quality that prevent these embarrassments. This paper presents a framework for estimating and measuring how improvements in application quality relate to business value. Projecting the actual business benefits an organization will achieve by improving the internal quality of an application requires a detailed knowledge of the business it serves

II. How Do You Measure the Internal Quality of a Business Application?

The internal quality of an application describes among other attributes the soundness of its architectural design and the extent to which its implementation follows proven coding best practices. Internal quality is not measured by passing test cases that were mostly designed to verify the functional correctness of an application. Rather, measures of internal quality assess the non-functional attributes of the application – *the internal structure and engineering of its code*.

Highlights

CAST has identified five key areas of internal software quality that most affect business outcomes.

Through extensive research and industrial experience CAST has identified five areas of internal software quality that most affect business outcomes. These five areas, or 'health factors', are similar but not identical to the high level software quality measures defined in ISO 9126. Each of these five areas can be assessed by measuring numerous attributes of the software, and then aggregating the results into a summary health factor for that area (see Table 1). These health factors summarize internal software quality at a level that can be related to business outcomes and value. These health factors and some of the business benefits they affect are summarized in Table 1.

Table 1 - Application Health Factors and Their Benefits to the Business

Health Factor	Description	Example business benefits
Transferability	*Attributes that allow new teams or members to quickly understand and work with an application*	• *Reduces inefficiency in transferring application work between teams* • *Reduces learning curves* • *Reduces lock-in to suppliers*
Changeability	*Attributes that make an application easier and quicker to modify*	• *Improves business agility in responding to markets or customers* • *Reduces cost of ownership by reducing modification effort*
Robustness	*Attributes that affect the stability of the application and the likelihood of introducing defects when modifying it*	• *Improves availability of the business function or service* • *Reduces risk of loss due to operational malfunction* • *Reduces cost of application ownership by reducing rework*
Performance	*Attributes that affect the performance of an application*	• *Reduces risk of losing customers from poor service or response* • *Improves productivity of those who use the application* • *Increases speed of making decisions and providing information* • *Improves ability to scale application to support business growth*
Security	*Attributes that affect an application's ability to prevent unauthorized intrusions*	• *Improves protection of competitive information-based assets* • *Reduces risk of loss in customer confidence or financial damages* • *Improves compliance with security-related standards and mandates*

Highlights

The internal quality of business applications is critical for achieving important IT outcomes, which in turn are critical for achieving key business objectives.

III. How Health Factors Affect Business Value

There are five primary business objectives against which the costs of an application's poor internal quality can be assessed. These business objectives include:

- *Reducing risks to the business*
- *Increasing business agility*
- *Improving business productivity*
- *Optimizing IT's contribution to the Business*
- *Improving customer experience*

The internal quality of business applications is critical for achieving important IT outcomes. These IT outcomes in turn affect a large number of business outcomes that are critical for achieving business objectives. This chain of value creation from application health factors to business objectives is depicted in the right to left flow of relationships in Figure 1. These relationships do not represent a one-to-one mapping since health factors and IT objectives interact in myriad ways with business outcomes and objectives.

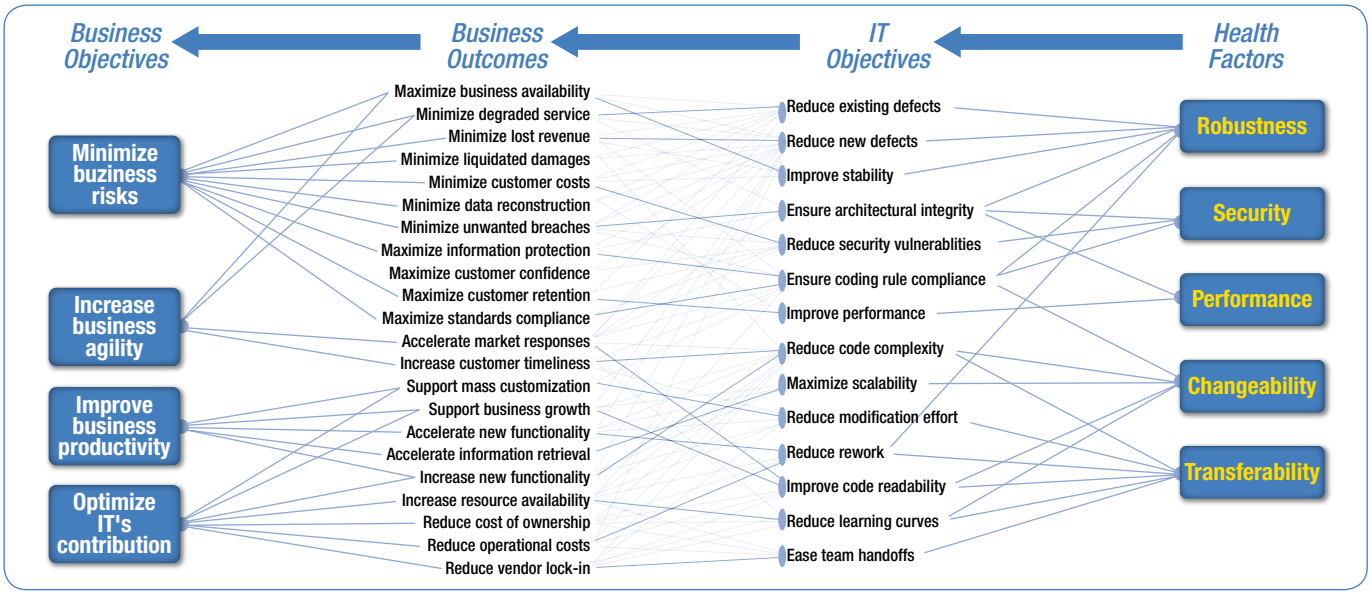

Figure 1. Relationships between health factors and business objectives

Highlights

We don't need sophisticated models to show the connection between application quality and business objectives. Simple equations will do.

Each business objective may be composed from a set of more specific objectives against which the value of application quality can be assessed. For instance, business risks come in several types:

- *outages in critical business systems,*
- *corrupted data,*
- *violated security, and*
- *regulatory noncompliance.*

Each of these more specific business objectives may have a unique set of business outcomes that affect it. IT outcomes will differ on which specific business outcomes they most affect. Figure 2 displays this more detailed analysis of how health factors affect of the four specific business objectives underlying the objective of reducing business risks.

Although we could represent these relationships with greater sophistication using systems dynamics, this diagram is adequate to show that the internal quality of an application has direct and wide-ranging impact on business outcomes and the achievement of business objectives. Even without building a quantitative model of the relationships in this diagram, it is possible to use simple equations that express how application quality affects the business value derived from an application.

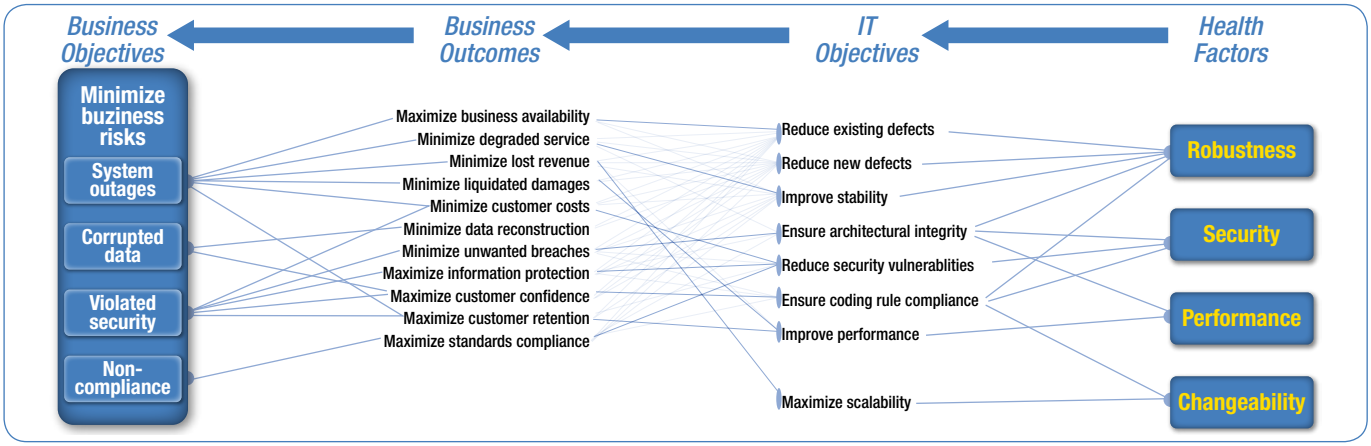

Figure 2. Relationships between Health Factors and Specific Business Risk Objectives

Highlights

The cost of internal quality problems can be expressed as the loss of known current or future revenue, the cost of underutilized workers, liquidated damages, and other costs based on historical business data.

The next sections will present equations that demonstrate how these relationships can be used to express the business value of the internal quality of applications. These equations are not presented as formal ROI analyses, since we do not include the cost of money, investment recovery periods, and other components of formal ROI models. Rather they represent loss functions attached to each business objective that indicate how poor internal quality is translated into lost revenue and increased costs.

IV. Reducing Risks to the Business

Business risks often present the easiest case for quantifying quality benefits. The cost of quality problems can be expressed as the loss of known current or future revenue, the cost of underutilized workers, liquidated damages, and other costs based on historical business data. Different types of risks experience different types of losses.

Outages that terminate business transactions – Consider a simplified evaluation of the loss due to an outage in a commercial application such as a reservation or customer order system. The costs involve lost revenue, effort for the business to recover and reactivate transactions, spikes in help desk and related costs for managing customer interactions, liquidated damages, and other costs that may be unique to the specific area of business. Although not all costs may be triggered in each outage, they should be considered to ensure they do not remain hidden in ongoing business activities.

$$\begin{aligned} Loss = {} & ((avg.\ revenue\ per\ minute) \times (number\ of\ minutes\ unavilable)) + \\ & cost\ to\ reactivate\ business + \\ & ((Additional\ customer\ service\ minutes) \times (\$\ per\ minute) + \\ & future\ revenue\ lost\ from\ defecting\ customers + \\ & liquidated\ damages\ if\ applicable + \\ & other\ related\ costs) \end{aligned}$$

Highlights

Although expensive, the identifiable costs of security breaches are often less than the lost revenue from customers who either no longer use the application or who defect to competitors they perceive to be more secure.

Corrupted data – The costs associated with corrupted data include recreating accurate data, redeveloping or correcting inaccurate outputs and reports, and any liabilities resulting from the use or reporting of inaccurate data. The cost of data corruption is compounded by the length of time before the inaccurate data is detected, since in many cases this makes the cleanup more extensive. The business bears many of the costs of data corruption either through having to correct the data or for unproductive downtime while IT restores accurate data.

Loss = cost of data reconstruction +
cost of recreating accurate reports +
liabilities created from inaccurate data +
other related costs

Violated security – Defects in an application's architecture or code often create vulnerabilities that hackers and cyber-criminals exploit to penetrate the system. These costs can include those related to theft of business or customer information, repairing malicious damage, informing customers of possible compromised information, security improvements to systems and business processes, liquidated damages, and lost future revenue from defecting customers. Although expensive, the identifiable costs of these breaches are often less than the lost revenue from customers who either no longer use the application or who defect to competitors they perceive to be more secure.

Loss = cost of stolen resources +
cost of rectifying data, records, or accounts +
cost of informing customers +
cost of security improvements +
future revenue lost from defecting customers +
liquidated damages if applicable

Highlights

The more confusing the interface, the more customer support personnel must be available to assist customers in completing transactions. Usability and responsiveness contribute to revenue growth and lower customer interaction costs.

Regulatory noncompliance – Weaknesses in an application's code can place the enterprise in non-compliance with industry standards or government regulations. While noncompliance can result in financial penalties, violations of regulations such as Sarbanes-Oxley can have criminal implications.

Loss = cost of penalties for noncompliance +
cost of bringing the system into compliance

V. Maximizing Customer Experience

As more customer transactions move online, business applications increasingly become the face of the business. The customer's experience with the company becomes their experience with the application's usability and performance. Confusing user interfaces, labyrinthine Websites, and glacially slow system responses frustrate customers. At a minimum, these problems reduce the amount of business a customer may transact, and in the worst case they drive customers to competitors.

To compound the customer loyalty problem, confused customers create even more costs per transaction. When customers have difficulty using a company's automated business systems, they call customer support to conduct transactions that could have been performed online. The more confusing the interface, the more customer support personnel must be available to assist customers in completing transactions. For instance, when system response is slow, customers often begin pushing buttons that may be interpreted as commands to pages that have yet to display. Many of these incorrect entries require staff time to undo and return to the original state in a customer's account. Usability and responsiveness contribute to revenue growth and lower customer interaction costs.

Loss = future revenue lost from departing customers +
future revenue lost from reduced transactions with loyal customers +
cost of conducting transactions shifted from online to customer service +
cost of customer service calls related to online difficulties

Highlights

The more needlessly complex the architecture or coding of an application, the longer it takes to add or modify functionality, verify its correctness, and deliver it into operation. Improving agility and reducing business risk produce immediate business benefits.

VI. Increasing Business Agility

Modern communication technology (cellular grids, the Internet, etc.) has multiplied the pace of business. Consequently, competition has shrunk the time available for responding to customer demands and market conditions. The agility of a business in responding quickly is strictly limited by the internal quality of its applications. The more needlessly complex the architecture or coding of an application, the longer it takes to add or modify functionality, verify its correctness, and deliver it into operation. Worse, unnecessary complexity induces more mistakes and rework lengthening the time to develop and transfer new functionality into operation.

Loss = revenue lost from missing the customer's buying window +
revenue lost to faster moving competitors +
future revenue lost from defecting customers +
diminished profit from dilution of first mover advantage +
diminished economy of scale from loss of market share

The value of internal software quality to business agility is in terms of lost opportunity cost. It represents the lost revenue or market share experienced when competitors can respond quicker or when the response misses the customer's buying window. The importance of business agility cannot be overstated for long term business viability and growth. Both improving agility and reducing business risk produce immediate business benefits. However, improvements in business agility also affect a company's ability to execute its business strategy and optimize long term results.

Highlights

Outages affect more than business revenue. They drain IT staff productivity and morale.

VII. Improving Business Productivity

Since a primary purpose of many applications is to improve business productivity, defects which cause outages or performance degradation rob the organization of the full benefit from its investment in both the application and the workforce using it. Business losses caused by outages were covered as business risks. The productivity impact on employees and revenue generation will be treated separately here. The costs of lost productivity are usually calculated in terms of additional personhours to complete work or of lost opportunity for revenue generation. Even if the additional hours were performed as unpaid overtime, there is an impact on staff morale that can further reduce productivity or induce voluntary turnover. In addition, reduced productivity can inject delays into the completion of tasks that have other costs such as penalties for late delivery.

***Loss** = [(1 – (reduced output under degraded performance ÷*
avg. output at normal performance))
x number of workers affected
x avg. hourly cost
x hours of degraded performance] +
[(1 – (reduced output or revenue under degraded performance ÷
avg. output or revenue at normal performance))
x performance
x avg. revenue per hour] +
costs of delayed work

VIII. Increasing IT's Contribution

For most large enterprises, IT is funded as a fixed percent of the corporate budget, typically from 4% to 4.5%. Since IT is a fixed percent in the budget, cost savings in IT usually translate into freed resources available for additional work, rather than as a return of funds to the business. The value proposition then becomes, "How can we get more business functionality for our fixed investment?"

Highlights

Studies have proven that 50% of maintenance work is devoted to trying to understand the code rather than adding functionality. Reductions in code complexity can reduce this understanding time so that more time can be spent developing valuable functionality.

Usually around half of IT's budget is spent on application development and maintenance. The internal quality of business applications controls a surprisingly large proportion of these costs and dictates how much of application development resource will be allocated. For instance, every hour spent in rework fixing quality problems is a lost opportunity to provide additional value to the business. Similarly, studies have proven that 50% of maintenance work is devoted to trying to understand the code rather than adding functionality. Reductions in code complexity can reduce this understanding time so that more time can be spent developing valuable functionality.

Loss = ((proportion of annual effort spent on rework) x (avg. burdened cost of a developer) x (number of developers)) + (((Proportion annual effort spent modifying or enhancing existing code) x (.5 proportion of maintenance spent understanding)) x (avg. burdened cost of a developer) x (number of developers))

IX. An Example of Business Loss Due to Poor Internal Quality

Using the formula for application outages that terminate business transactions, consider the business costs of a 1-hour outage for an application that yields $120 in revenue per minute ($7200 per hour). In addition to lost revenue, the business will spend $1000 in employee time to verify, correct, or regenerate partially completed transactions and to verify the system is working correctly when brought back online. The help desk experienced a surge of 250 additional minutes of customer service calls at $2 per minute. Business intelligence analyses discovered that 20 existing customers made no further contact with the company after the outage, with annual revenues from those customers being $150 per year. Fortunately this application did not involve any liquidated damages. This total cost of this outage is estimated to be:

Loss = ((60 minutes) x ($120 per minute)) + $1000 reactivation cost + $500 customer service costs + $3000 lost customer revenue this year

Loss = $11,700

Highlights

A 10% decline in application performance can quickly add up to almost half a million dollars of business productivity loss in a quarter.

If the company experienced a 1 hour outage with this application once per quarter, the total annual loss to the business just from outages would be $46,800, not including IT costs for repairing the cause of failure, retesting the application, and similar IT costs. These IT costs will be treated as rework in a subsequent analysis since they reduce IT's ability to contribute to the business.

Next consider that in addition to outages this application's performance degrades by 10% due to poor database access procedures that reduce its ability to handle an increasing load of business transactions. Also consider that this application supports 100 knowledge workers whose fully burdened cost is $25 per hour. This loss of productivity is calculated both as lost revenue and as loss in the productive value of knowledge worker compensation. The cost of this application's internal quality problems per quarter is estimated to be:

Loss = *((0.1 output) x (100 workers) x ($25 per hour)*
*x (500 hours per quarter))**
((0.1 output) x ($7200 revenue per hour) x (500 hours per quarter))
Loss = *$485,000 per quarter*

This number may be reduced by voluntary unpaid overtime to get work completed. Nevertheless, poor internal quality is still robbing the organization of a substantial portion on its expected return from the investment in this application and its workforce.

Finally consider the cost of enhancing and maintaining this application. Consider there are 5 developers assigned full time to this application with an annual fully burdened cost of $100,000 each. Approximately 35% of their time is spent fixing defects, while 50% is spent maintaining and enhancing existing code. The estimated lost contribution from investment in IT by the business is:

Loss = *((.35 proportion of rework) x ($100,000 burdened rate)*
x (5 developers)) +
(((.5 annual effort spent on modification) x (.5 proportion of maintenance spent understanding)) x ($100,000 burdened rate)
x (5 developers))
Loss = *$300,000*

Highlights

When the value of software quality is computed only against development or maintenance costs, the full impact of the loss remains invisible to the business. However, when computed against business costs and lost business opportunities, a strong business case can be made for investing in quality.

The loss from the risk of outages, lost business productivity, and the lost contribution from IT attributed to poor quality, is surprisingly large. Based on the sizes of these losses, improvements in the quality of applications offer substantial benefits to the business. When the value of software quality is computed only against development or maintenance costs fully contained within IT, the full impact of loss due to poor quality remains invisible to the business. However, when computed against business costs and lost opportunities, it is much easier to make a strong case for investing in quality.

X. The Value for Improving the Internal Quality of Applications

Improving the quality of applications has two components; external, functional quality and internal, non-functional quality. Most defect detection and related quality activities built into standard application development and maintenance processes focus on external or functional quality. These are the defects that represent deviations from the specified requirements for the application. Advances in testing, peer review, and requirements management processes and technology have improved the ability of application development teams to detect and remove the majority of these defects before placing the application into operation.

Since internal, non-functional quality problems are harder to detect, hidden as they are in the architecture and engineering of the application, they are frequently the causes of outages, degraded performance, security breaches, corrupted data, and similar problems. These quality problems come in a wide range of manifestations, from bad coding techniques, to needlessly complex designs, to violations of coding standards. By detecting these internal quality problems and correcting those with the most critical priorities, application developers can dramatically increase the value of an application to the business.

Returning to our example application in the last section, correcting a non-functional defect that would cause a 1-hour outage will save the business $11,700 per outage, more if the outage were to last longer than an hour. Now consider the productivity impact of removing internal quality problems that reduce the degraded performance from 10% of its original capacity down to only 7% of original capacity. The impact would save the business $145,500 per quarter from the original loss of $485,000 (figure 3).

Highlights

When internal quality improvements reduce rework by 25% a year and enable maintenance staff to spend 60% less time making sense of the code, it frees up $75,000 of effort per application for creating new business functionality.

Loss = ((0.07 output) x (100 workers) x ($25 per hour)
x (500 hours per quarter)) +
((0.07 output) x ($7200 revenue per hour) x (500 hours per quarter))
Loss = $339,500 per quarter

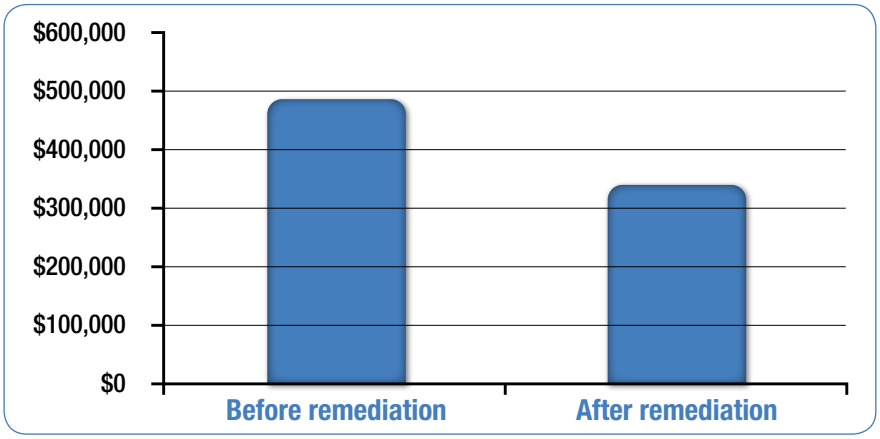

Figure 3. Example reduction in quarterly lost productivity from improving application performance

There are two ways that improvements in internal quality can affect the contribution of IT to the business. If improvements in the internal quality of this application were to reduce the proportion of annual effort absorbed in rework to 25%, then the team could contribute $50,000 more effort to producing new functionality for the business. Similarly, if improvements to the internal quality of the application reduced its complexity and developers spent only 40% of their modification/enhancement time understanding the code, they could contribute $25,000 more effort to producing new functionality for the business. As shown below, the amount of lost contribution to the business has been reduced to $225,000 with improvements to internal quality – freeing $75,000 of effort to contribute more value to the business.

Loss = ((.25 proportion of rework) x ($100,000 burdened rate)
x (5 developers)) +
(((.5 annual effort spent on modification) x .4 proportion of
maintenance spent understanding) x ($100,000 burdened rate)
x (5 developers))
Loss = $225,000

Highlights

By proactively detecting internal quality problems and correcting them, IT teams can dramatically increase the business value of an application.

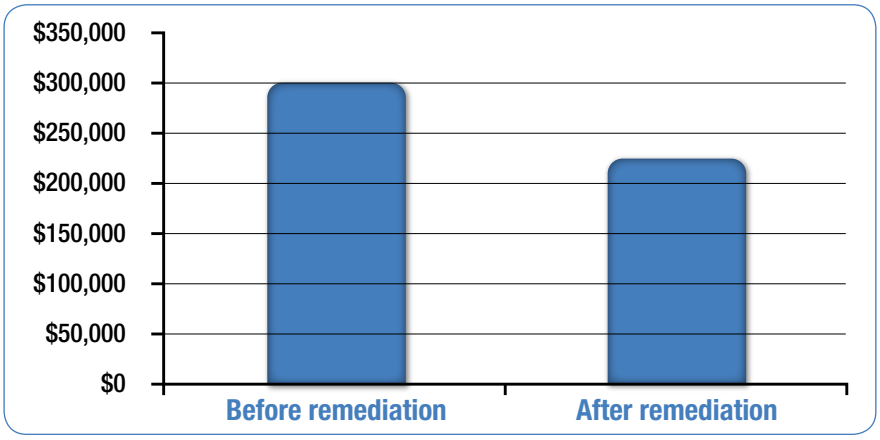

Figure 4. Example reduced quarterly loss in IT contribution to the business after improving internal application quality

Compared to the licensing, installation, learning, and usage costs of automated tools for identifying internal quality problems, the ROI for internal quality improvements is substantial. Amounts will differ by the size and related attributes of the application, but for business-critical applications the value to the business of improvements is dramatic and long-lasting. IT organizations must craft these types of analyses to defend their investments in improving the internal quality of applications.

Bill Curtis is an industry luminary who is responsible for influencing CAST's scientific and strategic direction, as well as helping CAST educate the IT market on the importance of managing and measuring the internal quality of its software. He is best known for leading the development of the Capability Maturity Model (CMM) which has become the global standard for evaluating the capability of software development organizations.

Prior to joining CAST, Dr. Curtis was a Co-Founder of TeraQuest, the global leader in CMM-based services, which was acquired by Borland. Prior to TeraQuest, he directed the Software Process Program at the Software Engineering Institute (SEI) at Carnegie Mellon University. Prior to the SEI he directed research on intelligent user interface technology and the software design process at MCC, the fifth generation computer research consortium in Austin, Texas. Before MCC he developed a software productivity and quality measurement system for ITT, managed research on software practices and metrics at GE Space Division, and taught statistics at the University of Washington.

Dr. Curtis holds a Ph.D. from Texas Christian University, an M.A. from the University of Texas, and a B.A. from Eckerd College. He was recently elected a Fellow of the Institute of Electrical and Electronics Engineers for his contributions to software process improvement and measurement.

Dr. Bill Curtis
Senior Vice President and Chief Scientist

About CAST

CAST's unique technology is the result of more than $70 million in R&D investment. Top engineering talent, dedicated to building the best technology for assessing the internal quality of mission-critical applications, has made CAST the leader in Automated Application Intelligence. CAST's mission is to enable the world's best enterprises to achieve significantly more business productivity from their complex IT software systems.

Founded in 1990, CAST has helped more than 650 organizations worldwide speed IT delivery to the business, mitigate risks in production, improve customer experience, and reduce the total cost of application ownership. CAST is listed on NYSE-Euronext (Euronext: CAS) and serves Global 2000 organizations worldwide with a global network of locations in the US and Europe.

www.castsoftware.com

CAST Headquarters
North America: +1 212-871-8330
Europe: +33 1 46 90 21 00