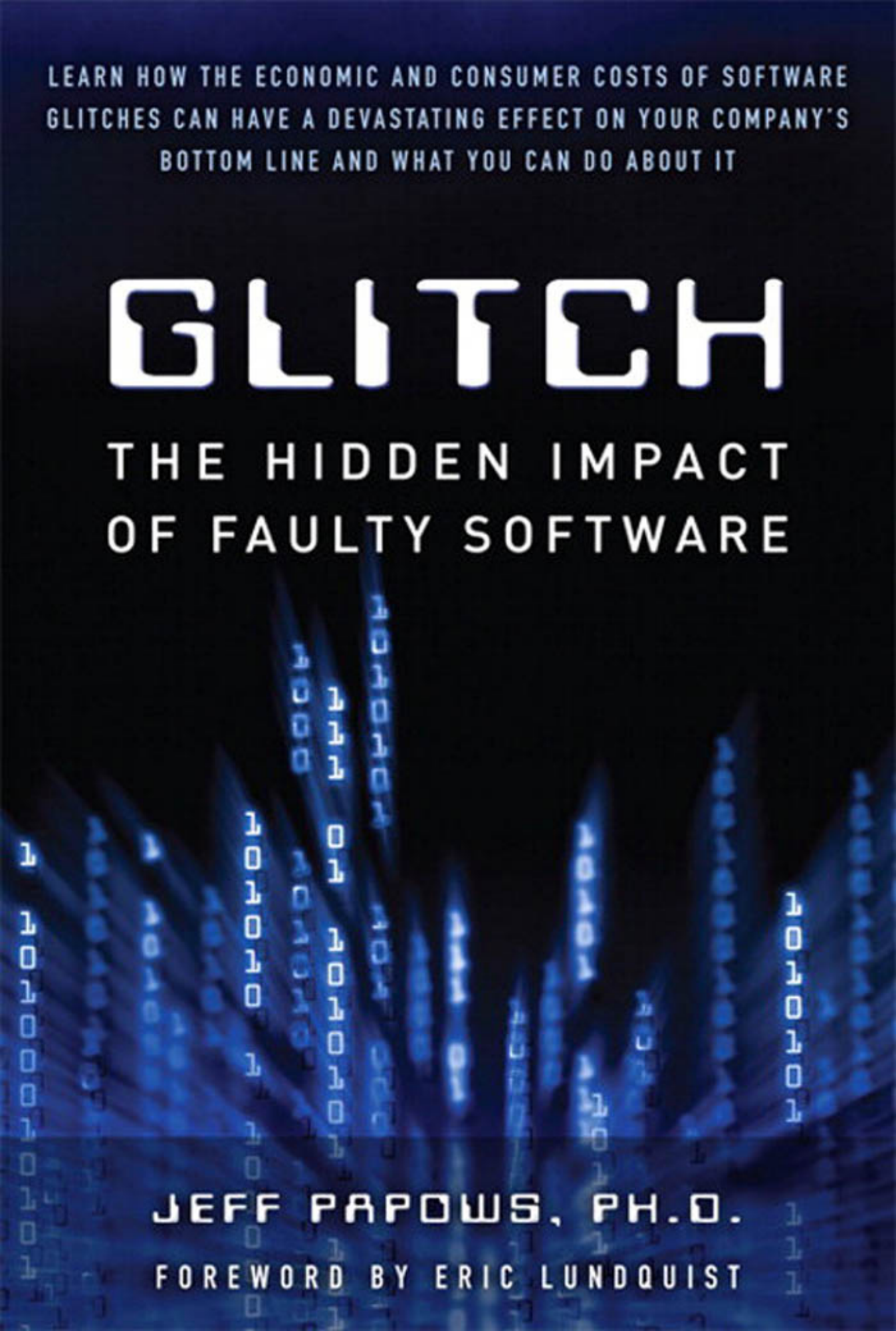
LEARN HOW THE ECONOMIC AND CONSUMER COSTS OF SOFTWARE GLITCHES CAN HAVE A DEVASTATING EFFECT ON YOUR COMPANY'S BOTTOM LINE AND WHAT YOU CAN DO ABOUT IT

GLITCH

THE HIDDEN IMPACT OF FAULTY SOFTWARE



JEFF PAPOWS, PH.D.

FOREWORD BY ERIC LUNDQUIST

Library of Congress Cataloging-in-Publication Data:

Papows, Jeff.
Glitch : the hidden impact of faulty software / Jeff Papows.
p. cm.
Includes bibliographical references.
ISBN 978-0-13-216063-6
1. Computer software--Quality control. 2. Manufactures--Defects. 3. Information technology--Management. I. Title.
QA76.76.Q35P36 2010
005--dc22

2010025053

ISBN-13: 978-0-132-16063-6
ISBN-10: 0-132-16063-3
Text printed in the United States on recycled paper at R.R. Donnelley in Crawfordsville, Indiana.
First printing September 2010

Associate Publisher
Greg Wiegand

Acquisitions Editor
Katherine Bull

Managing Editor
Kristy Hart

Project Editor
Jovana San Nicolas-Shirley

Copy Editor
Gayle Johnson

Indexer
Lisa Stumpf

Proofreader
Water Crest Publishing

Publishing Coordinator
Cindy Teeters

Cover Designer
Alan Clements

Compositor
Nonie Ratcliff

Table of Contents

Foreword

Glitch might be too kind a word for too big a problem. In the technology industry, a glitch can be a simple, short, unnoticed interruption in a network connection or a file that gets saved in the wrong place. However, a glitch can also be as serious as life and death. The question isn't whether we need a new word to cover the span from short annoyances to major technology meltdowns, but how to avoid the glitch that can put you and your company in peril. That is the issue Jeff Papows, Ph.D., sets out to solve in this book.

The list of glitches Papows covers in this book is truly staggering. A driver in a serious car accident is able to dial 911, but a glitch prevents the nearest fire department from being alerted to the emergency. A motorist stops to buy a pack of cigarettes and finds a charge for $23 quadrillion and change on his bank statement. What started as a simple software upgrade at a nuclear power plant spirals into an automatic, unplanned shutdown.

These and many other glitch stories are part of this book, but a simple list of glitches would not help solve this very big problem. Where Papows distinguishes himself is in writing a book that does not simply accept computer and technology problems as an unalterable happenstance, but as the end result of a faulty set of business and technology practices. It was the idea that computer shutdowns and other assorted malfunctions are not just part of doing business that attracted me to Papows' manuscript and why I was happy to be asked to write this Foreword.

In my course of covering the technology industry as a journalist for over twenty years, I've had the opportunity to write a lot about computer glitches, bugs, and major meltdowns, from the ILOVEYYOU worm in 2000 that coursed through Windows systems via email to the cascade of business process and technology failures that prevented BP

from detecting and preventing the disastrous oil rig blowout currently devastating the Gulf of Mexico. Too often, in my opinion, these and other technology meltdowns are seen as an act of inevitable happenstance at best or a dire, but acceptable, consequence of technology advance, which is a regrettable but necessary part of scientific progress. Papows shows both those scenarios to be wrongheaded and a result of faulty thinking rather than faulty electronic widgets or software programs.

What this book accomplishes is to provide a guide to business and technology managers wanting not only to root out and prevent glitches before they strangle a company's financial life, but also allowing technology advances to improve our society. Creating a society without the fear of a technology-based disaster unfolding just one glitch away would be a remarkable and noble achievement.

Although, as Papows explains in detail, there is not a simple "anti-glitch" piece of electronic wizardry to prevent future technology malfunctions, there are a set of predefined skills and business processes a company can wield to protect itself against headline-grabbing technology meltdowns. One of the greatest contributions of this book is to call on technology educators and industry to think big and redefine the roles of the software engineer, restructure information technology governance, and create business processes where technology is used to accelerate an idea into a product or service offered to the public without a company killing lurking glitches unseen. One of the strongest pieces of advice that Papows offers is to embrace mistakes early and often in the technology development process as a way to squash those minor bugs before they grow to the size of business-killing problems. The steps he offers in killing those minor bugs before they become major issues don't require advanced technology degrees or enormous capital investments, but do require that all parties speak a common business language, have a common set of goals, and discard finger-pointing blame sessions in favor of producing a

bug-free product or service. It is in explaining how to create that bug-free environment that Papows offers a unique and important contribution to business and government leaders.

The timing of this book could not be more important. As a society, we sit aghast as we watch a digital video feed of spilled oil fouling major fisheries and pristine beaches. Meanwhile, as our personal and business lives become ever more intertwined in online social networks, and vital business operations now take place in a computing cloud, business and individual life depend on technology operations to run flawlessly 24 hours a day. Preventing glitches from disrupting or destroying our digital-dependent society is what this book is all about, and it's worth your time to read and understand Papows prescription to keep those nasty computer bugs at bay.

Eric Lundquist
Vice President, Strategic Content
Ziff Davis Enterprise

CHAPTER 2
The Personal Impact

Glitches have become quite commonplace in headlines and in our personal lives. We usually don't pay attention to them or are no longer surprised when they happen—unless it is something so massive and dangerous that it disrupts our lives.

Because we are pouring more technology into automobiles and medical devices, it is not a stretch to say that glitches can sometimes be a matter of life and death. This chapter explores the issues surrounding Toyota vehicle recalls, as well as the impact of faulty technology on radiation machines designed to help treat cancer patients.

From there, we'll address the role that consumers, business leaders, and government officials can and should take to help reduce the impact of these life-threatening computing errors.

Toyota: From Class Act to Class Action

Toyota, the world's largest auto manufacturer,[1] is one company whose glitches have been front and center. Once it was a symbol of quality and safety. But Toyota's reputation took a nosedive when the company was forced to recall vehicles as news of deaths and injuries to drivers made headlines around the world. This bad publicity was most prevalent throughout the latter half of 2009 and the first half of 2010.

On January 21, Toyota announced the first in a series of product recalls that would occur throughout 2010, and prompt government action. The first voluntary recall of the year was

for 2.3 million vehicles across eight of its brands that were manufactured between 2005 and 2010. The recall was due to accelerator pedals that may mechanically stick in a partially depressed position or return slowly to the idle position.[2] Essentially, Toyota issued the recall to warn owners that the vehicles may accelerate or decelerate on their own. Five days later, Toyota suspended the sales of the potentially affected models.[3]

On January 27, dealing with a separate issue related to accelerator defects, Toyota sent a letter to the United States National Highway Traffic Safety Administration (NHTSA). In the letter, Toyota amended its Defect Information Report that was filed on October 5, 2009, stating the potential risk for floor mat entrapment of accelerator pedals in certain Toyota and Lexus models.[4] What could potentially happen in the instances outlined in the recall is that the accelerator pedal gets trapped in the floor mat and continues to increase the vehicle's speed while diminishing the driver's ability to control the automobile. As a side note, it is the auto manufacturer's legal responsibility to alert the NHTSA within five days of discovering a product defect.

Pedal entrapment is exactly what happened to the Saylor family of Chula Vista, California. Mark Saylor, his wife Cloefe, their 13-year-old daughter, Mahala, and Cloefe's brother, Chris Lastrella, were on their way to Mahala's soccer practice in a Lexus ES350 on August 28, 2009. When the car's accelerator got caught in the floor mat, Mark Saylor couldn't control the vehicle as it quickly accelerated to over 100 miles per hour. The car went through an intersection on a dead-end road, sideswiped another car, crashed through a fence, landed in a riverbed, and burst into flames. Unfortunately, there were no survivors.[5] The tragedy of the Saylor family was one of many incidents involving Toyota vehicles; the majority of reports cited problems with the car's accelerator.

The issues for Toyota escalated after the United States Department of Transportation received several complaints about braking difficulties in the 2010 Toyota Prius hybrids.

This led to the February 4, 2010 opening of an investigation into Toyota by the Department of Transportation.[6] Four days later, Toyota announced a voluntary safety recall on approximately 133,000 2010 model year Prius vehicles and 14,400 Lexus Division 2010 HS 250h vehicles so that Toyota could update the software in the antilock brake system (ABS).

According to the formal statement issued by Toyota, "Some 2010 model year Prius and 2010 HS 250h owners have reported experiencing inconsistent brake feel during slow and steady application of brakes on rough or slick road surfaces when the ABS (antilock brake system) is activated in an effort to maintain tire traction."[7]

The bottom line with the Prius recall is an issue with the software.[8] The Toyota recalls continued across the company's various brands due to additional mechanical issues that were categorized as glitches until a fuller investigation could be conducted.

The U.S. Government Gets in the Driver's Seat

Toyota's successive product recalls—more than eight million vehicles in 2010—led to fuller investigations by the U.S. government, including the U.S. Department of Transportation[9] and the U.S. House Committee on Oversight and Government Reform.[10]

The prepared testimony delivered by Toyota President and CEO Akio Toyoda echoes the issues that are facing many companies today, not just auto manufacturers. In his statement before the U.S. House Committee on Oversight and Government Reform, he said, "Toyota has, for the past few years, been expanding its business rapidly. Quite frankly, I fear the pace at which we have grown may have been too quick." He added, "We pursued growth over the speed at which we were able to develop our people and our organization, and we should sincerely be mindful of that."[11]

Like many companies that are in the midst of continued growth, it's easy to lose sight of the fundamentals that are

baked into the technology and are the catalyst for that growth. Anecdotally, if you've ever been through a downsizing, you've likely heard the mantras about getting back to basics and focusing on what matters. If we could sustain that mind-set regardless of fluctuations in the economy, we might see less technology-related catastrophes that result from failing to focus on the right things.

The massive Toyota recalls prompted the U.S. House Energy and Commerce Committee to propose to Congress the Motor Vehicle Safety Act of 2010.[12] From a technology point of view, the bill suggests several improvements to how vehicles are designed, engineered, tested, and manufactured. It also makes provisions for the inclusion of "event data recorders" that will be included in every automobile starting in 2012. These event data recorders are a scaled-down version of airplane black boxes. They are designed to help provide more accurate reporting in the event of a crash or air bag deployment.

Financial Implications

Toyota is facing hefty government fines, along with recall costs and lawsuits. Not the least of these line items was the $16.375 million fine imposed by NHTSA, the maximum fine allowed, for failure to notify it of the pedal defect for almost four months.[13]

Toyota's final tally from these glitches has yet to be determined, although estimates range from $3 to $5 billion. The actual costs will vary, depending on class-action lawsuits that include death and serious-injury claims. Also, deeper investigations will occur into previous accidents that may have erroneously been categorized as driver error as opposed to gas pedal malfunction. These are just the tip of the iceberg for Toyota when you think about the impact of automobile resale value, car dealers' bottom lines, insurers that paid claims where Toyota was ultimately responsible, and so on.

However, let's not be fooled into thinking that the issues at Toyota are isolated and are not part of the larger, industry-wide

technology issues that are looming. The overwhelming public concern is quite valid, and Toyota has issued subsequent apologies and updates to show how it's addressing the problems. However, I suspect that Toyota won't be the only auto manufacturer to face such a public flogging because of software glitches.

Lessons Learned from Toyota

As more automobiles are instrumented with technology, it's important to keep the lessons learned from Toyota top of mind. Three critical lessons can be learned from this situation:

- Be forthcoming about potential product issues, even if they haven't yet resulted in injury. Contributing to Toyota's image problem as well as the financial toll was Toyota's delayed response to the accelerator issue.
- Success and continued company growth need to be carefully managed and aligned with technology processes that are focused on the customer. This is especially true with manufacturing products that can affect a consumer's quality of life.
- We need a more effective way of testing and introducing new technology into automobiles. Just as you need a license to drive, I propose that we apply that same principle to the engineers who design and develop technology. We could require a stringent technology licensing, certification, and renewal process for IT governance in the automobile industry.

The Technology Behind the Wheel

The technology that's included in automobiles these days, such as global positioning systems, keyless entry, and parking assistance, is brilliant. As much as we like to think that embedding technology in automobiles is a relatively new idea, it's been happening for decades, for better and for worse.

Based on data from the NHTSA, since the introduction of technology into vehicles 30 years ago, the number of electronic system recalls in the U.S. has tripled.[14] This isn't surprising considering that IT analysts at Frost & Sullivan report that a modern luxury car contains close to 100 million lines of software code. Who'd have thought that much technology would be required to pick up a gallon of milk and a loaf of bread?

Considering that we've become accustomed to having our appliances, computers, and devices fully loaded, it only makes sense that we apply those same wants and needs to our vehicles.

The advances in automotive engineering and design as well as IT will only continue to increase the amount of technology we embed in vehicles. A look into the future reveals that we've only just begun to explore the inclusion of massive amounts of technology in our automobiles.

We may have adjusted to the idea of allowing a DVD player in our vehicles to occupy the kids on long rides, but are we ready to allow the Internet into our cars? We'd better be if the analysts at market research firm iSuppli are correct in their prediction that by 2016, 62.3 million global consumers will have Internet access in their cars.[15] Figure 2.1 illustrates the expected growth of Internet-connected cars.

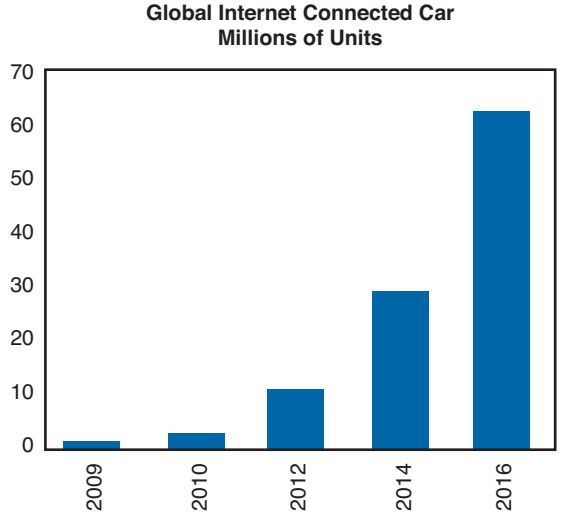


Figure 2.1 *Global Internet-connected cars 2009–2016.*

The safety implications of having an Internet browser on our automobile dashboards present their own set of issues. This rings especially true when you consider that in 2008 nearly 6,000 people died and more than half a million people were injured in crashes involving a distracted driver.[16]

I suspect automobile manufacturers will put legally approved warnings in place to protect them from the fallout that's likely to come when you allow car owners to simultaneously drive and surf. Yet these types of innovations call into question whether we are using technology to add value on behalf of the consumer or simply doing it because we can.

Although distracted drivers are not directly linked to glitches, what you have to remember is that the addition of more technology into an automobile—even to ease the driving experience—can increase the propensity of glitches.

Due Diligence for Enterprise Software Procurement

As a culture, we're inundated with marketing messages designed to convince us that the latest and greatest widget will change our lives, solve our business problems, make us smarter, and transform us overnight. The IT industry is no exception. Because much of the enterprise-class software that we're talking about is complex, distinguishing the buzzwords from the actual business value that the product delivers is not always easy.

Even after thorough product testing, evaluations, and what's known as proof of concept (POC), which puts the product through its paces in the customer's real-world environment, mixed or failed results can occur after significant financial and intellectual investments have been made.

To help filter quality products from the latest marketing campaigns, the following seven criteria should be part of the due diligence process:

- *Think like a customer.* You often hear people in the IT industry talk about aligning technology with business goals. Although this is important, the customer should

be the priority. In a globally connected world, competition can come from anywhere, and loyalty is rooted in the quality of service that the customer receives. If the conversation you're having with the IT vendor goes down the road of how the software can do wild and crazy things like streamline business processes, ask the vendor how this benefits your customer. For every feature and benefit that is pitched, respond with questions about the value to the customer.

- *Don't just buy; invest.* Consider the decision to buy technology an investment, not a static purchase, because the technology will continue to evolve and improve just as your company does. For enterprise software, you can expect the value of the investment to become clear within 18 to 24 months. For technology that is more consumer-oriented in nature, such as subscription-based tax preparation software, the same principles apply, although the return on investment is more immediate. In both scenarios, the customer is investing in the vendor's technology because it has proven value and is far more economical than hiring a team. The longer-term investment pays off in the form of efficiency and productivity that will increase through continued use of the technology.
- *Justify the cost.* The cost justification for the technology purchase comes down to simple economics. The formula for determining whether the investment is worthwhile is based on the organization's staff and skill set. Most software vendors have created their own return-on-investment (ROI) calculators as part of the sales process. Although these are a good starting point for determining whether the investment is worthwhile, they should not be taken at face value. One formula is to multiply the cost of hiring a team of software engineers (E) by the cost (C) of the software and divide that by the amount of time (T) required to realize ROI:

$$E \times C \ / \ T = ROI$$

To factor in the cost of hiring staff, keep in mind that the average salary for an application software engineer as of May 2008 was $85,430, with the highest 10 percent of this population earning more than $128,870.[17] When it comes to entry-level positions, the average starting salary offer for graduates with a bachelor's degree in computer science averaged $61,205 in 2009.[18]

- *Evaluate the vendor.* Equally important as evaluating the software itself, if not more important, is to consider the health of the vendor that's selling it. Research the track records of the engineering team, the founder, and the executive team. You want to be sure that if you invest in the technology, the vendor will be around in the future to continue supporting you. This doesn't mean you should consider only the major software vendors when it comes to purchasing decisions. Many smaller, niche players can serve specific business needs that may be underserved by the larger players. In this instance, explore how the company sustains itself. Is it bootstrapped, funded by angel investors, or backed by established venture capitalists (VCs)? If so, who's behind the money, and what is their track record?
- *Determine the product's actual version number.* You'll almost never find a version 1.0 of any product. The industry is well aware that the 1.0 label signifies that it's the first time the product is being released, which likely means that all the kinks have not been worked out. This is why you'll often find products that start with version 3.0. This doesn't mean the product is faulty or that a version 3.0 isn't just that. However, the version number is something to fully explore with regard to how the product will actually work after it's installed.
- *Ask for customer references.* This may sound like a no-brainer given the time and costs associated with making a technology purchasing decision. Tread carefully down this path. Be leery of a vendor that claims to have impres-

sive customer references, but the customers' corporate policies won't allow them to talk. Although this may be true from a public relations perspective, a satisfied customer should be available to speak to a prospect off the record. When you do get to that conversation with the customer, be sure to ask how long they've been using the product, if they receive a discount for being a reference, and the specifics of the product's best and worst features.

- *Study industry analyst reports.* There are mixed reviews in the IT industry regarding the unbiased evaluations conducted by the analyst community. In many cases, analysts are a valuable resource to help companies determine their technology needs and which vendors are most capable of addressing them. They also provide in-depth market reports and forecasts. However, this community has a dark side that I'd argue is steeped in the analysts' preferences for vendors that subscribe to their services. The analyst community shouldn't be overlooked when it comes to evaluating technology, but you should ask if the vendors they are recommending to you are also their clients.

The Road Ahead

There's a lot to think about when technology is added to automobiles as well as other infrastructures and devices without a system of governance to ensure the quality of the products that are supposedly being enhanced. Specifically, I'm talking about IT governance. This includes a set of processes, policies, and best practices that are used to ensure that the best possible "glitch-free" software code is used as the foundation for nearly all our technology innovations.

Technology folks, especially those at the managerial level, are familiar with the term IT governance, which could help address many of these glitches. But to be clear, especially

because you'll be reading more about it, I want to underscore that IT governance as it relates to glitches is not the same as compliance. I mention this because many people use these terms interchangeably. IT governance is complementary to the branch of technology called compliance that made its way into the spotlight as a result of the passing of the Sarbanes-Oxley Act of 2002.

Yet saying that IT governance is important and actually making it a reality are two very different things in many organizations. According to Lynn Cox, IT program manager at Ford Motor Company, "You have to educate the developers on the importance of IT governance. You can require mandatory training, but sometimes people will just show up and not pay attention. What you need to do is make it real for them. Share stories of real things that happen because of a lack of IT governance."[19]

Of course, Cox wouldn't disparage Ford's competition, but I would venture a guess that real-world stories and the role of IT governance are discussed more often at Ford these days in light of the Toyota situation.

Taking the Pulse on Healthcare IT

When it comes to healthcare and medicine, technology continues to play a critical role. Perhaps you were able to head off major dental surgery because your dentist took X-rays that revealed issues that had not yet risen to the surface. Or perhaps your child received x-rays when he fell off a swing. These common preventive measures can be quite helpful in quickly diagnosing breaks and fractures and avoiding potentially painful treatments down the line. Yet all of this adds up to a sevenfold increase in a person's average lifetime dose of diagnostic radiation since 1980.[20]

According to a series of articles on radiation that appeared in *The New York Times*, it has become woefully apparent that glitches are making their way into the very treatments that are supposed to save our lives. Included as an appendix in this

book is one of the articles in the series, "Radiation Offers New Cures, and Ways to Do Harm."[21] It spells out the impact of these software glitches and their role in the deaths of several patients. A synopsis of the article follows.

Synopsis of the Article "Radiation Offers New Cures, and Ways to Do Harm"

Unless you really know your way around an oncology ward, you probably aren't familiar with a linear accelerator, or Linac. Essentially, this device is used to treat cancer patients by delivering a uniform dose of radiation to specifically treat a tumor. The beams that are delivered through the Linac destroy cancer cells while sparing the surrounding healthy tissue.

On the plus side, newer technology in Linac allows doctors to more accurately attack tumors and reduce certain mistakes. As with many computer-centric activities, there is a culturally accepted mind-set that because the process is computerized, it can't be wrong. Medicine is one area where that perception and the complexity curve collide. On the negative side, the complexity has created more opportunities for glitches to occur in terms of software flaws and faulty programming. These types of glitches impact the delivery of X-ray beams, as many patients have unfortunately discovered.

One of those patients was Scott Jerome-Parks. Before one of his radiation treatments for tongue cancer, Nina Kalach, the medical physicist responsible for overseeing the Linac, input the dosage and patient information into the software application. Kalach's input into the system would determine how much radiation the Linac would administer.

When Kalach tried to save her work, the computer froze. It's important to note that the software and Linac,

provided by Varian Medical Systems, require three essential programming instructions that must be saved in sequence. The first step is the dose of radiation in the beam, the second is a digital image of the treatment area, and the third is the instructions that guide the multileaf collimator. This is a device within the Linac that is made up of individual "leaves" of high atomic numbered material that can move in and out of the path of a particle beam to block it from hitting unintended areas in the body with radiation.

Before the software program aborted, Kalach received an error message asking if she wanted to save her changes, and she replied yes. At that point, the system rebooted, and Kalach believed her changes were saved. Later that day, the computer crashed again and was again rebooted.

Six minutes after the second reboot, Jerome-Parks received the first of three radiation treatments. The next day he had another dose, as was the prescribed course of action. After the second dose, it was apparent from Jerome-Parks' physical condition that something had gone horribly wrong. His head and neck were swollen almost beyond recognition, and he was writhing in pain.

Nevertheless, Jerome-Parks underwent a third dose of radiation. Since the evidence was mounting that the patient was having more than an adverse reaction to the treatment, Kalach conducted a test on the technology and discovered that the multileaf collimator, which was supposed to focus the beam precisely on the tumor, was wide open. This meant that not only had Jerome-Parks' entire neck, from the base of his skull to his larynx, been mistakenly exposed, but he also had received seven times his prescribed dosage of radiation. Kalach also later learned that the software changes related to the patient's data were never saved before the computer crashed.

After his radiation treatments, Jerome-Parks continued to suffer from acute radiation toxicity. He could barely sleep or swallow, and he was hiccupping and vomiting. He needed a feeding tube and a constant stream of drugs and supplements. As his illness got worse, Jerome-Parks lost his hearing, eyesight, and balance. He died of acute radiation poisoning at the age of 43.

According to reports from the hospital that treated Jerome-Parks, similar system crashes "are not uncommon with the Varian software and these issues have been communicated to Varian on numerous occasions."

Varian's president and chief executive officer, Timothy Guertin, stated that the company had distributed new software with a fail-safe provision and also had warned customers to be especially careful when using their equipment.

Unfortunately, that updated software didn't arrive in time to help a woman who, several months later, was being radiated for cancer of the larynx. In this particular case, therapists tried to save a file on Varian equipment when the system's computer screen froze. Again, the multileaf collimator was wide open, and this particular patient received nearly six times her prescribed dose.

On the same day that warnings were issued to hospitals regarding Linac and its related software in light of the Jerome-Parks case, Alexandra Jn-Charles, 32, started radiation treatments for breast cancer. After 27 days of treatment, it was discovered that the Linac was missing a filter.

This resulted in Jn-Charles receiving three times the prescribed amount of radiation. It also resulted in a gaping wound in her chest that would not heal and eventually created a hole that exposed her ribs. After the radiation, Jn-Charles was repeatedly hospitalized for pain and had to live with the odor that was coming from the wound.

During this time, her cancer returned. Several months after her wound had finally healed, Jn-Charles passed away.

The stories of Scott Jerome-Parks and Alexandra Jn-Charles are not isolated incidents. A Philadelphia hospital gave the wrong radiation dose to more than 90 patients with prostate cancer and kept quiet about it. Meanwhile, in 2005, a Florida hospital disclosed that 77 brain cancer patients received 50 percent more radiation than prescribed because the linear accelerators had been programmed incorrectly for nearly a year. In another report about radiation missteps, one patient with stomach cancer was treated for prostate cancer, and another patient with brain cancer received radiation treatment intended for breast cancer.

Where Technology and Human Intellect Intersect

In fairness, it's important to note that not all of these mistakes were solely the result of technology. In several instances, human errors such as poor safety procedures or inadequate staffing and training also played a part.

What's more important to acknowledge is that the details of the radiation cases just discussed are shielded from public view by the government, doctors, and hospitals. Although privacy is a major concern, it seems that a bit more disclosure is needed, at least within the medical community, to help avoid these issues in the future. Moreover, no single agency oversees medical radiation. Therefore, accidents are underreported—if they are reported at all—because this isn't a requirement in all states. Realizing the potential problems associated with this issue, the New York State Legislature, along with the hospital industry, agreed in the 1980s to report medical mistakes. However, the identity of the institutions that made the mistakes remains cloaked.

Where is the line between human error at the hands of the Linac machine and at the hands of the keyboard when the software code is being written? Is it realistic to expect radiation physicists to become experts in computer programming, and vice versa? Just how much training goes into ensuring that hospital staff have mastered the use of the technology? How can software developers create more error-free programs?

These are complex issues and certainly can't be solved within the confines of this book. Besides, addressing these issues crosses many lines in technology, medicine, and government. However, I raise the questions to hopefully prompt discussions that will perhaps lead to awareness and action among those who can effect change. At the end of this chapter, I include suggestions for how we can more effectively address these issues as a society.

If you were wondering why I emphasize the importance of IT governance, these medical stories clearly underscore my reasoning. It is critical that software developers fully understand the impact of their efforts and the role that IT governance must play in the design and development of software.

I suspect the *The New York Times* report sparked many discussions at dinner tables and throughout the healthcare industry, many of which were centered on the likelihood of radiation poisoning happening to them or a loved one. According to Bill Klein, principal at Noblis Health Innovation's National Recall Center, "Over 48 percent of radiology recalls concerned software, with hardware problems following up at 38 percent."[22]

Noblis is a nonprofit science, technology, and strategy organization that is widely known for its RASMAS National Recall Center service. RASMAS helps healthcare facilities track recalled and defective supplies and equipment in 15 different product domains, including biologics, blood products, toys, food, pharmaceuticals, radiology products, and tissue.

What's a consumer to do? While the responsibility rests on the software developer, manufacturer, doctor, and techni-

cian, Klein suggests that patients ask their physicians about maintenance procedures, equipment operation, and staff accreditation. Additionally, patients can ask technicians and staff if there are established procedures to ensure that safety notices are dealt with quickly.

I also believe that patients and their families can use the power of technology for good. There's no reason why we can't stir a movement online to create a dedicated, comprehensive website that educates and informs the public about good and bad service at hospitals. If we can rank our experience at a hair salon or pizza parlor, why can't we take these social media technology tools a step further to warn people about potentially life-threatening experiences at local hospitals?

Lying by Omission

The issues at Toyota and those that *The New York Times* article brings to light are complex in that a variety of factors are associated with those tragic deaths and injuries. Aside from questioning the IT governance that was or wasn't in place at the software vendor, many other actions and people can be called into question. These include but are certainly not limited to the radiation physicists, hospital administrators, and government officials who are not demanding more stringent reporting of radiation poisoning.

Yet when it comes to full disclosure regarding these glitches, when is the boy crying wolf and unnecessarily alerting consumers to hazardous products, and when is it okay to delay notifying the public? Surely a more proactive approach to identifying and mitigating the risks associated with these glitches is the more strategic, cost-effective, and potentially life-saving course of action.

As IT and business professionals, we can no longer tolerate obfuscation of these glitches in the automotive and healthcare industries until they are discovered by consumers or required by law to become a matter of public record. We need to lead the charge to initiate the IT Governance Manifesto.

Not all healthcare glitches are as extreme as those outlined here. However, in an effort to improve the patient experience through technology, sometimes the best intentions go awry and wind up costing far more than anticipated.

Oregon: A Lesson Learned in Healthcare IT

In 2008, the Oregon Health Payment Plan was transitioning to a new $80 million IT system. Incidentally, the U.S. federal government is covering 90 percent of the costs of this system, which processes $200 million worth of claims each month.[23]

After two false starts, the system finally went online in December of that year. By September 2009, it had yet to accurately enroll and track residents who were eligible for services. For example, a report written in the morning might have indicated that a person was enrolled, while a report written in the afternoon said the opposite. Due to this glitch, the state of Oregon estimates that 2,800 new patients were "misplaced" over the course of a year, representing a loss of $9 million in annual revenue. Meanwhile, an Oregonian managed-care organization believes it has paid pharmacy and emergency room bills for patients that may not have been enrolled.

When the errors in the Oregon system dragged on for over nine months, the deputy director of Human Services hand-delivered a letter to contractor Electronic Data Systems (EDS). The letter demanded that the problems be fixed within 90 days, or the state could file suit. Twelve months of ongoing IT issues directly affected the bottom line for the state of Oregon, its healthcare providers, its residents, and the federal government. The actual cost is hard to quantify, because when they were asked about the financial impact, state officials and healthcare providers said they didn't know the answer.

Acknowledging the complexities involved, officials at the Oregon Department of Human Services said they expected glitches given the scope of the project.[24] I believe that somewhere in the middle of this mess lies a more balanced ground

between the complexity of the system and the complacent attitude that errors will occur.

The issue in Oregon is yet another example illustrating that the scale of our infrastructures, the pace at which productivity must continue in IT, and the underlying economic factors are colliding despite our best efforts.

The issues at the Oregon Health Payment Plan are not unlike the IT projects that are currently under discussion or under way throughout the healthcare industry.

Throwing Good IT Dollars After Bad

While the Internet on the car dashboard is an obvious example, there are many instances in which technology is introduced into an infrastructure with the best intentions and worst execution.

Software errors are inevitable and glitches are unavoidable to a certain extent, but we should not invest more technology into a problem without a full understanding of the fundamental issues that initially caused the problem. Before any IT purchasing decisions are made, companies should undertake an extensive due diligence process.

The IT Governance Manifesto

Imagine if we could make consumers more aware of the potential risks lurking inside a product, system, or infrastructure. We've seen it with cigarettes and alcohol, but we have yet to see similar warnings applied to technology.

Companies would balk at the idea of having to publicly admit to shortcomings in their products. However, a third-party warning system is worth considering when it comes to products that affect our health and safety. I suggest this because I strongly believe and also gravely fear that we will see a rise in the number of software glitches before serious steps are

taken to reduce their occurrence and the overall impact of glitches that manage to sneak past inspection.

The groundswell of personal health and safety issues due to software glitches will give rise to yet another dramatic consumer-driven market shift that will force change upon businesses of every size and in most industries.

The shift that's under way reminds me of President John F. Kennedy's Consumer Bill of Rights that was introduced in 1962.[25] Kennedy was responding to consumers demanding increased rights and legal protection against bad business practices. Kennedy's speech outlined six basic rights: The Right to Be Safe, The Right to Choose Freely, The Right to Be Heard, The Right to Be Informed, the Right to Education, and the Right to Service.

With this in mind, I firmly believe that consumers and businesses need to lobby government to pass legislation that mandates higher standards and establishes more concrete pass/fail criteria to eliminate the gray areas that so many products fall into. Product recalls are not enough.

This is why I'm proposing the IT Governance Manifesto. Making this vision a reality will require a cross-section of IT and business professionals, government agencies, and consumer advocacy groups that will join to accomplish the following:

- Lobby for new legislation that requires more stringent reporting of software glitches in matters of life and death.
- Impose fines on individuals and organizations responsible for software glitch cover-ups that put consumers' health and/or safety at risk.
- Require a specified level of IT governance at organizations that produce products that can directly affect a consumer's quality of life.

We can't sit idly by until the next auto or medical device manufacturer becomes the source of our personal tragedy or the

subject of a government investigation. The expediency of the Motor Vehicle Safety Act of 2010 is evidence of how quickly the government can move when consumer safety is at stake. Therefore, there's no reason why we can't collectively start lobbying for the IT Governance Manifesto.

Endnotes

1. World's largest auto manufacturer rankings. The International Organization of Motor Vehicle Manufacturers. http://oica.net/wp-content/uploads/world-ranking-2008.pdf.
2. Toyota press release. Toyota Files Voluntary Safety Recall on Select Toyota Division Vehicles for Sticking Accelerator Pedal. http://pressroom.toyota.com/pr/tms/toyota/toyota-files-voluntary-safety-152979.aspx.
3. Toyota press release. Toyota Temporarily Suspends Sales of Selected Vehicles. http://pressroom.toyota.com/pr/tms/toyota/toyota-temporarily-suspends-sales-153126.aspx.
4. Toyota press release. Toyota Amends Recall on Potential Floor Mat Interference with Accelerator Pedal. http://pressroom.toyota.com/pr/tms/toyota/toyota-amends-recall-on-potential-153214.aspx.
5. AOL Autos. "Lexus Crash: An Avoidable Tragedy." Reilly Brennan. December 10, 2009. http://autos.aol.com/article/toyota-tragedy-saylor-family/.
6. U.S. Department of Transportation. NHTSA Launches Probe into Timeliness of Three Toyota Recalls. http://www.dot.gov/affairs/2010/dot2910.htm.
7. Toyota press release. Toyota Announces Voluntary Recall on 2010 Model-Year Prius and 2010 Lexus HS 250h Vehicles to Update ABS Software. http://pressroom.toyota.com/pr/tms/toyota-2010-prius-abs-recall-153614.aspx?ncid=12067.
8. Ibid.
9. National Highway Traffic and Safety Administration. United States Department of Transportation Responds to Third Toyota Recall. http://www.nhtsa.gov/PR/DOT-25-10.
10. United States House Committee on Oversight and Government Reform. "Toyota Gas Pedals: Is the Public at Risk?" February 24, 2010. http://oversight.house.gov/index.php?option=com_content&task=view&id=4798&Itemid=2.

11. Prepared Testimony of Akio Toyoda, President, Toyota Motor Corpora. tion. Committee on Oversight and Government Reform. February 24, 2010. http://oversight.house.gov/images/stories/Hearings/Committee_on_Oversight/2010/022410_Toyota/TESTIMONY-Toyoda.pdf.
12. U.S. House Energy and Commerce Committee. Motor Vehicle Safety Act of 2010. April 29, 2010. http://energycommerce.house.gov/Press_111/20100429/NHTSA.Motor.Vehicle.Safety.Act.Discussion.Draft.Text.pdf.
13. National Highway Traffic Safety Administration. Statement from U.S. Transportation Secretary Ray LaHood on Toyota's Agreement to Pay Maximum Civil Penalty. http://www.nhtsa.gov/PR/DOT-71-10.
14. The number of automobile recalls has tripled: National Highway Traffic Safety Administration databases. http://www-nrd.nhtsa.dot.gov/Cats/index.aspx; Over 100 million lines of software code in luxury vehicles: Frost and Sullivan. "Convergence in North American Auto Industry." December 15, 2008.
15. iSuppli. Automotive Research Topical Report. "Internet in the Car: The Future of In-Vehicle Connectivity." Q4 2009.
16. United States Department of Transportation. Distraction.gov: Official U.S. Government Website for Distracted Driving. National Highway Traffic Safety Administration. http://www.distraction.gov/stats-and-facts/.
17. U.S. Bureau of Labor Statistics. Occupational Employment Statistics (OES) Survey Program. May 2008 National, State, Metropolitan, and Nonmetropolitan Area Occupational Employment and Wage Estimates.
18. Winter 2010 Salary Survey. National Association of Colleges and Employers.
19. Interview with Lynn Cox by Kathleen Keating. December 2009.
20. *New York Times* series: "The Radiation Boom. When Treatment Goes Awry." "Radiation Offers New Cures, and Ways to Do Harm." Walt Bogdanich. January 23, 2010.
21. Ibid.
22. Interview: Bill Klein and Kathleen Keating. February 2010.
23. Oregon Department of Human Services, Division of Medical Assistance Programs. January 11, 2010. http://www.oregon.gov/OHA/OHPB/meetings/2010/care-or-1012.pdf.
24. Oregon Live. "Glitches Plague State Data System." August 7, 2009.
25. John F. Kennedy. Address to U.S. Congress. March 15, 1962. John F. Kennedy Presidential Library and Museum.

Index

D

E

F

G

H

I

J

K

L

M

N

O

P

Q

R

S

T

U

V

W