A New Approach to System Verification

Lessons Learned from Other Industries

White Paper

Abstract

System verification approaches for high-tech equipment have not scaled in step with increasing technical complexity. While software developers now enjoy integrated development environments (IDEs) that offload mundane programming tasks, QA teams still spend much of their time performing repetitive tasks such as configuring test beds, executing regression tests manually, documenting existing test cases, and attempting to reproduce bugs. Meanwhile, the creative testing that could radically improve quality remains undone. This white paper suggests a new approach to system verification derived from the successes of other industries facing comparable business challenges.

Fanfare

Table of Contents

Introduction

For equipment manufacturers, success in the current business climate hinges on two main factors: fast time to market and customer satisfaction. Quality Assurance (QA) plays a major role in both goals. Today's rising equipment complexity is forcing some manufacturers to feel that they must choose between missing market opportunities or shipping products without adequate verification. The evidence of that choice is today's high field failure rates and declining customer satisfaction.

The underlying problem is that system verification tools and processes have not scaled in step with increasing equipment complexity. While business application developers now enjoy integrated development environments (IDEs) that offload them from mundane programming tasks, verification and QA teams still spend much of their time performing repetitive tasks such as configuring test beds, executing regression tests manually, documenting existing test cases, and attempting to reproduce bugs. Meanwhile, the creative testing that could radically improve quality remains undone.

This white paper suggests a new approach to system verification derived from the successes of other industries facing similar business challenges.

Why a new approach to system verification is needed

High-tech equipment validation processes, little changed over the years, are overdue for an update. Several factors have converged to drive change.

Exponential rise in equipment complexity

Moore's Law, which predicts a doubling of the density of transistors on integrated circuits roughly every 18 months, continues to hold true. More sophisticated hardware enables equipment manufacturers to continually add more and better product features to differentiate their products. Even more complexity arises from the fact that many manufacturers now offer a wider variety of devices, and that more of these are embedded or integrated, requiring verification of operation with new protocols, middleware, and applications.

Faster time to market

Average time to market from concept to delivery for high-tech equipment has shrunk from five or six years in the '90s to just one or two years today—or even less. The first manufacturer to introduce a high quality product stands to gain a potentially sustainable competitive edge. As companies race to introduce more features in less time, the controlling factor for product introduction has shifted from the speed of software development to the speed of validating that the software works. In fact, according to industry estimates, system verification consumes an average of 50% of development time and 25% of costs for high-tech equipment. As a result, some vendors' product introduction cycles are actually lengthening.

Globally distributed development and verification

A trend toward globally distributed system verification complicates coordination of product releases. When the development and system verification teams work in different countries and speak different languages, they lack a standard way to describe defects and test results, creating delays and misunderstandings that can affect field quality.

Less sophisticated equipment users

Traditional consumers of sophisticated equipment have been network administrators with the training to competently troubleshoot, upgrade firmware, and navigate complex documentation and support websites. Today, many high-tech manufacturers have expanded their market to also include home users, who purchase equipment such as broadband modems, wireless routers, and networked digital video recorders (DVRs) for their televisions. Support lines are clogged with calls from people less able to perform troubleshooting on their own, adding a new source of costs.

Lack of commercially available system verification tools

Although verification and QA typically consume as much as 50% of the development lifecycle, the high-tech equipment industry still lacks commercially available, automated verification tools for equipment software. Consequences include:

- Steady degradation of field quality
- Declining customer satisfaction
- Slower time to market, reducing revenue windows

The historical reason for the lack of commercially available tools was that early high-tech equipment was controlled exclusively via custom interfaces. Today, devices are configured, controlled, and monitored through a variety of standard mechanisms including command line interfaces, embedded Web interfaces, or other Application Programming Interfaces (APIs) such as a Tcl API. This standardization has enabled the development of commercial automation tools

What high-tech equipment manufacturers can learn from other industries

To increase customer satisfaction, equipment manufacturers need a breakthrough approach that leapfrogs today's incremental quality improvements. Inspiration for such an approach can be found in the successes of other industries that have overcome similar challenges: application software development, civil engineering, chip design, and manufacturing.

Use the right tool for the job

Application software development

Today's application software developers are developing richer user interfaces in less time than ever before. The enabler is integrated development environments (IDEs), which combine the editor, compiler, and other tools in the same interface. Programmers now need far less time to debug because they can see error messages and the code associated with the error at the same time. Visual C, for example, helped developers design higher quality Windows application, more quickly. Similarly, the move from vi to emacs significantly boosted productivity for UNIX developers.

Lessons learned. Creative people can be more creative—and more effective—with tools that automate mundane aspects of their jobs. Like the best software developers, the best testers are creative people, whose gift is considering possibilities and making them happen. The difference is simply the nature of the possibilities they deal in.

Verify often—not just at the end

Application software development: modern programming philosophies

More inspiration comes from modern programming movements such as Agile Development and Extreme Programming. The Agile Manifesto (www.agilemanifesto.org), for example, advocates incremental, collaborative development—that is, testing each small change as it is developed in order to more quickly identify and fix defects.

Lessons learned. Conducting system verification in parallel with development creates efficiencies. When less code changes between tests, developers spend less time hunting for bugs.

Assign high value to quality

Civil engineering

Civil engineers do not design beautiful structures and only afterwards consider their integrity. Rather, quality is inextricable from every design and building decision. What's at stake is more than company reputation and customer satisfaction, but actual lives: Whether the building stands or the levee holds is not an afterthought, but the primary goal.

Lessons learned. Civil engineers pay attention to quality as if people's lives depend on it—because they do. Certain types of high-technology equipment are becoming nearly as vital. The importance of quality is obvious for medical devices and airplanes, and is becoming more so for networking equipment. Public safety agencies depend on network availability and high performance to access the voice, video, and data they need to gain situational awareness and plan an effective response. Similarly, all levels of government rely on the network for information-sharing and collaboration during day-to-day operations as well as disasters. Attention to quality can no longer be casual: the stakes have grown too high.

Compensate for increased complexity by designing at a higher level

Chip design

During the '80s, the complexity of chips increased by a factor of 100. It would be reasonable to expect longer time to market. And yet, the average number of iterations of a design to bring a chip to market actually dropped from three to about one-and-one-half, and time to market decreased from two or three years to less than one.

Lessons learned. The semiconductor industry adopted tools and processes to design chips correctly, the first time. Their approach: separate chip design from the underlying process. Designers now use a hardware development language (HDL) to describe the desired behavior of the circuits, leaving the actual transistor layouts to a silicon compiler. A simulation tool is used to verify the circuit's operation before it is built. Separating design from process improves quality, accelerates development, and relieves designers from having to be experts about underlying processes so that they can focus on their core expertise.

Aim high—but be pragmatic

Manufacturing

When Japan introduced the concept of zero-defect manufacturing in the '70s, the notion was widely regarded as naïve at best. Western manufacturers believed it far more cost-effective to simply moderate the number of defects. They reconsidered when, in 1977, the Shizuoka plant of Matsushita Electric's Washing Machine Division achieved continuous zero defects over several months in a drain-pipe assembly line operation. Just one decade after the concept was introduced, Six Sigma had become a standard manufacturing model. The Six Sigma approach employs a rigorous process methodology that tolerates a defect rate of 3.4 defects per million (.0003%). When defects are discovered, a team of engineers analyzes them to discover the root cause and implement process changes to prevent the defects from ever occurring again. This increased scrutiny improves manufacturing processes without increasing overall costs because Six Sigma manufacturers no longer need to absorb the cost of so many defective goods and the associated support burden.

Lessons learned. Six Sigma manufacturers do not approach quality by tackling every defect at the same time and refusing to ship product until all defects have been addressed. Rather, they use Pareto lists to assign priority to defects according to their impact on total product quality, and work on them in order of importance. Quality begets more quality because when defects decrease, companies can afford to devote teams to discovering the root cause of rarer problems.

Five lessons for equipment software QA

Manufacturers of high-technology equipment can apply lessons learned from other industries to their own quest for higher quality and greater profitability:

- **Empower testers by automating mundane, error-prone, time-consuming activities**. The rote aspects of system verification siphon time from the creative aspects. Give QA teams the tools to automate tasks like configuring test beds and executing regression tests. This frees them to do their jobs better—and gives companies an edge in recruiting and retaining the best QA engineers.
  Source: Application development industry
- **Adopt parallel development and verification processes rather than serial processes.** Require developers and testers to collaborate as a team from day one, with the goal of developing not just a functional feature, but one that is fully verified. Interleaving verification with development makes it easier to fix bugs by narrowing the scope of inquiry.
  Inspiration: Agile Programming movement
- **Make quality a central element of the equipment design philosophy**. Treat it as if lives depend on it—because they might.
  Inspiration: Civil engineering
- **Take advantage of tools to operate at a higher level.** Use simulation tools to save time, enabling more rigorous verification and verification.
  Inspiration: Chip design industry's adoption of HDL
- **Be a pragmatist, not a purist.** Rather than battling all defects at once, follow the example of Six Sigma manufacturing companies, using Pareto lists to correct defects according to their impact.
  Inspiration: Six Sigma manufacturing initiatives

Summary

The successes of other industries demonstrate that breakthroughs in quality begin with a change in attitude. By increasing expectations for system verification—and providing testers with the tools to meet those expectations—high-tech equipment manufacturers position themselves to significantly increase quality, improve customer satisfaction, and gain a competitive edge.

About Fanfare

Fanfare is revolutionizing how high-tech equipment vendors manage product release cycles. By combining test development, execution, documentation, and automation into a single step, the company offers the first commercially-developed test automation software for validating the functionality of high-tech equipment. Using the FanfareSVT family of products, quality assurance teams improve customer satisfaction and reduce QA costs. Founded in 2004, Fanfare is a private company based in Mountain View, California, with venture backing from Matrix Partners and Redpoint Ventures.

For more information

To learn more about Fanfare and the FanfareSVT family of products, visit **www.fanfaregroup.com.**

Fanfare

Rev0606