

## The Quality of Your Code is now the Quality of Your Brand

By John J. Paliotta, Chief Technology Officer, Vector Software

### Introduction

Successful automotive companies must constantly innovate by introducing new technology to differentiate and improve their brands in today's intensely competitive environment. As a result, modern vehicles have evolved into complex integrated technology platforms with embedded software powering all major systems including: engine control, power train, braking and airbags, driver assistance, and infotainment.

These statistics illustrate the concept well: today's vehicles have more processing power than any of NASA's early spacecraft, including the Apollo lunar lander<sup>1</sup>. A modern vehicle has more lines of code (10M) than the latest technology in the US Joint Strike Fighter<sup>2</sup>.

Consumer opinions are influenced by their interaction with this technology. In many ways the brand is now the user experience, more so than the technical specifications.

### Technology Defines Brand

According to Interbrand's 15th annual Best Global Brands Report, a focus on integrated technology is helping to drive brand loyalty and value for automakers. Three out of the five "Top Risers" listed were from the automotive sector, making the recent annual report a record-breaking one for the auto industry. The collective value of the automotive brands appearing on the Global Brands ranking also increased 14.6 percent.<sup>3</sup>



Figure 1: Embedded software powers all major systems of today's automobiles including: engine control, power train, braking and airbags, driver assistance, infotainment and much more.

As automobiles evolve from mechanical to software-enabled devices, automakers must rethink fundamental product development principles, including moving from a sequential design process to a more agile approach, with higher degrees of collaboration between self-directed, cross-functional

<sup>1</sup> <http://www.physics.org/facts/apollo-really.asp>

<sup>2</sup> <http://www.informationisbeautiful.net/visualizations/million-lines-of-code/>

<sup>3</sup> [Interbrand's 15th Annual Best Global Brands Report](#), October 9, 2014

teams.<sup>4</sup>

### Software Quality Must Be a Top Priority

Statistics show that more than 50 percent of auto recalls are now due to software bugs, not mechanical issues<sup>5</sup>. With an industry average of 1 bug per thousand lines of code, the average car has 10,000 latent bugs. Software quality needs to be at the top of the list for auto brands looking to preserve – and elevate – brand status.



Figure 2: The dashboard in the connected car is similar to a smartphone, making them essentially a "tablet on wheels."

### Rethinking the role of Testing in the Development Cycle

In the 1950's, a revolution in manufacturing quality was initiated by W. Edwards Deming, who taught companies how to produce higher quality products at the lowest cost possible. Central to Deming's philosophy was his criticism of the dominant form of quality control in use at the time: products were inspected for defects only *after* they were made. In contrast, Deming maintained it was better to design the manufacturing process to ensure that quality products were created from the start.<sup>6</sup>

This seems obvious to us now when we think of manufacturing, but the software industry has been slow to adopt a similar philosophy. Often, the release cycle is seen as a zero sum game, where investing more time in testing simply pushes out release dates by the same amount. This mindset neglects the value of these test artifacts over the lifecycle of the product. This flawed thinking is deeply ingrained in the industry.

The first step to improved quality is to rethink the role of testing in the software development cycle. Organizations searching for improved quality must start thinking in terms of total lifecycle cost, not

---

<sup>4</sup> [The Connected Vehicle Movement](#), Booz Allen Hamilton, October 2014

<sup>5</sup> [The connected car of the future: Get ready for more software bugs](#), GeekWire, Jan 5, 2015

<sup>6</sup> [W. Edwards Deming of Powell, Wyo.: The Man Who Helped Shape the World](#)

simply time to release 1.0. Most organizations have a huge quantity of legacy tests, but often they are not being leveraged properly.

The following are common flaws in the implementation of testing:

- Testing completeness is not measured
- Tests are not easily shared between team members
- Testing is not automated and continuous

### ***Measuring Testing Completeness***

Source code coverage analysis is an easy-to-implement way to measure the completeness of existing tests. Measuring code coverage for all types of testing and combining this into a single view provides valuable insight into under-tested and over-tested areas of the application. By understanding gaps in existing tests, users are able to improve testing efficiency, and over time, fill these gaps with new tests. The result is improved application quality.

### ***Sharing Tests across Teams***

Tests are often compartmentalized, with the majority of tests being owned by the Software Quality Assurance (SQA) team. If developers do not have access to these tests, then it is not surprising that bugs are found by SQA late in the release cycle -- *after* new features are integrated and these tests get run for the first time. The lack of test sharing can also result in an adversarial relationship between developers and SQA as much time is wasted by developers trying to understand bugs, and SQA trying to diagnose test failures. Sharing tests offers a variety of benefits to the whole team:

- Bugs will be fixed on the day they are introduced, not weeks later
- Code changes that break existing tests will not be integrated
- Team members will focus on edge cases versus nominal path testing
- SQA will have more time to write new tests

### ***Automated and Continuous Testing***

The key to creating a sharable testing infrastructure is to ensure that tests are small and run fast, execution is automated and dependable, and results are easy to understand.

Automated and continuous testing will only work if a common testing platform is implemented across the organization that allows all team members to run existing tests and integrate new tests easily. The tests must capture all preconditions, test values, and expected results so that test execution is as simple as clicking a button and viewing a pass or fail status; and most importantly it must be easy for developers to debug failing tests.

### **Conclusion**

Over the last 50 years, automotive suppliers have revamped design and production processes to control cost, drive quality, and create brand loyalty. These gains may be lost if the same focus on quality is not applied to the software that controls the majority of systems in modern automobiles.