

Vector Software

WHITEPAPER

Change-Based Testing

Best Practices for Improving Testing Efficiency and Software Quality

According to recent Cambridge University research, the global cost of debugging software has risen to \$312 billion annually¹. This number tells us that the software industry has a serious quality problem, and that companies are spending incredible amounts of money treating the symptom, but not nearly enough remedying the cause. A major cause of software bugs is inefficient and incomplete testing. This paper will focus on techniques that can be used to improve testing efficiency.

Most software groups have hundreds or thousands of tests, so you can argue that before more tests are built, they should ensure that existing tests are being used to their best advantage. An obvious thing is to ensure that developers can run every test on every software change prior to integration.

This seems like a simple idea, but in reality few development groups have the ability to do this, because:

1. Running all tests takes a long time.
2. Many tests are complicated to run.
3. Test results are difficult to interpret.

Of these three problems, the most challenging one to solve is the length of time that it takes to run all tests; this is the primary obstacle to shortened release cycles and improved software quality.

Change-Based Testing Reduces Test Time

Change-Based Testing (CBT) helps organizations test faster and smarter by analyzing each code change against all existing test cases, and choosing the sub-set of tests that are affected by the change. By running only this sub-set of tests, test execution times are greatly reduced, and developers get immediate feedback on the impact of their changes. This allows bugs to be fixed immediately, when they are introduced, rather than weeks later, during “full” testing.

¹ Brady, Fiorenza, [Cambridge University Study States Software Bugs Cost Economy \\$312 Billion Per Year](#), PR Web, March 26, 2014

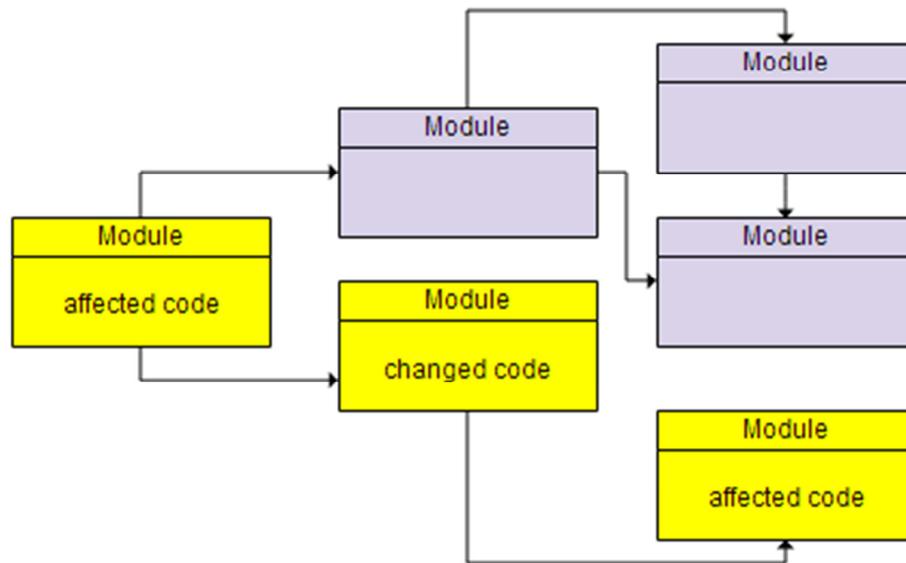


Figure 1: CBT identifies the direct and indirect effects of each code change

Change-Based Testing Improves QA Efficiency

Historically, “full testing” is run periodically after hundreds of source changes have been made, this results in lots of wasted QA time tracking down the root cause of test regressions. Implementing CBT keeps the main source baseline clean since all existing tests must pass before any changes are integrated. This allows Integration and QA teams to concentrate on building tests to validate new functionality, rather than diagnosing regressions in existing functionality. Organizations using CBT will more easily meet their release deadlines and produce higher quality products.

Change-Based Testing Improves Infrastructure Utilization

In addition to providing faster test times, CBT provides better utilization of testing infrastructure, including target hardware. By reducing the total tests that are run for each software change, more test cycles can be run per day, while still identifying regressions caused by individual changes. This is particularly powerful when test equipment and target hardware is in short supply.

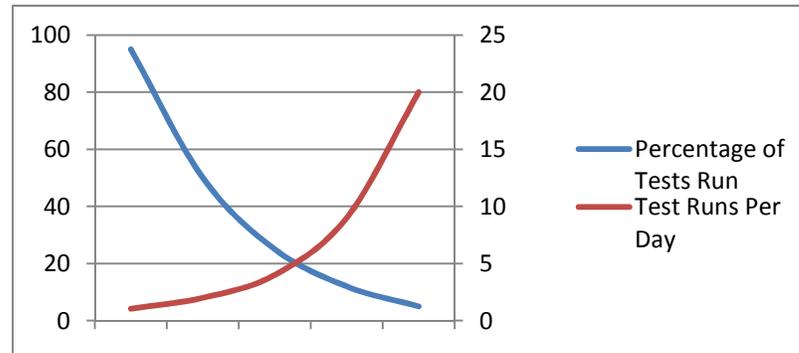


Figure 2: Change-Based Testing allows more test runs per day.

Change-Based Testing Improves Time to Market

Today's software development organizations are under tremendous pressure to complete complex development projects and meet narrow time-to-market windows.

Change-Based Testing Improves Quality

All of the above points are valid in isolation, but when considered together, they make a compelling case for Change-Based Testing as enabling technology for improved software quality.

The Business Case for Improving Quality

Much has been written about the quality revolution, which rolled across manufacturing over the past three decades. Curiously, these same proven concepts have not been applied to software. In fact, products shipped with faulty software are so prevalent that consumers have learned to simply power cycle products to solve "odd" behavior.

Software has become more than just a hidden component within products; in many ways it has become the component that differentiates products in crowded marketplaces. Additionally consumers are more empowered than at any time in history, via the web and social media tools, to provide instant feedback on products. This means that products shipped with buggy software damage the brand and a company's carefully crafted reputation.

VectorCAST and Change-Based Testing

All of the above points are valid in isolation, but when considered together, they make a compelling case for Change-Based Testing as enabling technology for improved software quality.

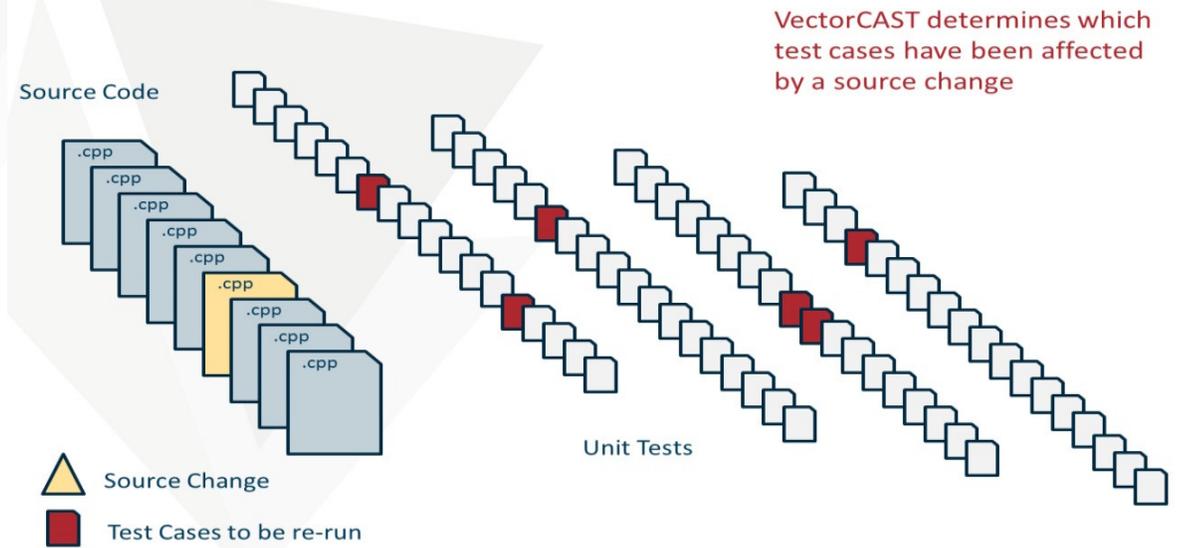


Figure 3: Change-Based Testing greatly reduces testing time while still ensuring testing completeness.

Conclusion

Solving the software quality challenge requires that companies invest in a testing infrastructure that enables their teams to produce high quality products with an efficient and reproducible process. The Change-Based Testing features of VectorCAST provide exactly this capability; allowing organizations to improve testing efficiency, which improves quality, reduces time-to-market, and increases customer satisfaction.

How much of the \$312 billion spent on software debugging is coming from your organization's budget?



About Vector Software

Vector Software, Inc., is the leading independent provider of automated software testing tools for developers of safety and business critical embedded applications. Companies worldwide in automotive, aerospace, medical devices, industrial controls, rail, and other industries, rely on Vector Software's VectorCAST™. By automating and managing the complex tasks associated with unit, integration, and system level testing, VectorCAST helps organizations accelerate the development and ensure the reliability of their embedded software applications.

Vector Software, Inc.
1351 South County
Trail, Suite 310
East Greenwich, RI 02818 USA
T: 401 398 7185
F: 401 398 7186
E: info@vectorcast.com

Vector Software
Golden Cross House
8 Duncannon Street
London WC2N4JF, UK
T: +44 203-603 0120
F: +44 77 969 11103
E: info@vectorcast.com

Vector Software
St. Töniser Str 2a
47906 Kempen Germany
T: +49 2152 8088808
F: +49 2152 8088888
E: info@vectorcast.com

Vector Software
Rm 403, Building 6, No.88
Daerwen Rd, Zhangjiang
Hi-tech Park Pudong New Area
Shanghai 201203 China
T: 21- 3126 8126
F: 21-5132 8526
E: info@apac.vectorcast.com