

Because developing for the embedded market is so complex, and getting more so every day, testing will always be a large, important task. Tools like Klocwork 2017 will continue to make this process less of a burden so developers can cut through the noise, fix what really matters, and move on to developing the next important part of a project.

TESTING OVERLOAD: WHERE DO WE GO FROM HERE?

With the recent decision by the Federal Aviation Administration that all Boeing 787 Dreamliner's must be reset every 21 days¹, it's no wonder that organizations are feeling the demands of testing — and testing efficiently. Many companies in the embedded market are already using tools like static code analysis (SCA) to help with testing, but with all the requirements and regulations you have to wonder: Have we reached testing overload?

The first real bug found in code was on September 9, 1945 in the Harvard Mark II². Computer scientists spent time over the next several decades writing, testing, debugging, and reviewing code — though not necessarily defined as individual tasks. It wasn't until the late seventies when debugging and testing were introduced as separate tasks³.

In 1979, Glenford Myers published his first version of a book, *The Art of Software Testing*, which helped establish how software development should be managed. Myers helped pave the way for cleaner coding, testing tools, and better software. But between 1979 and 2013, Myers' book has been through 69 editions. With that number — not to mention the many, many other testing books and guides that have been added into the market since — it's no wonder people feel overwhelmed.

Testing used to be done as an individual task, rereading code and correcting mistakes that could be found with the naked eye. There was also peer-review; better than self-reviewed code, but certainly still overwhelming and something that was destined to let issues fall through the cracks.

Flash forward to today and the tools available to help developers with testing are endless. And like books on proper testing, the same issue remains: How do you weed through all the options and know what tools are right for you?

Many developers in the embedded market use SCA as a testing tool while their creating code. SCA is effective because developers can find code issues and fix them before they're submitted into the entire code base, making it easier for multiple people on a project to work together. But SCA tools can have their

¹ ["Federal Register Volume 81, Number 232,"](#) US Department of Transportation. December 2, 2016.

² ["Moth in the machine: Debugging the origins of 'bug,'"](#) Computerworld. September 3, 2011.

³ ["Software testing history,"](#) eXtreme software testing.

own issues and their own barriers to success. For example, developers are often left with a very long list of issues that need to be corrected and it can be hard to figure out where to start and what issues are the most important.

It's a developer's dream to see defects in a prioritized list. Imagine how much easier testing would be if developers could triage defects depending on which were most likely to be a false positive, helping better manage their time and workload by avoiding time wasted on items that were incorrectly flagged as defects.

Luckily, embedded developers can now take advantage of a long-overdue ranking system for false-positive detection within SCA. Developers can cut through the noise and see exactly where teams should start with their testing efforts.

Klocwork 2017, from Rogue Wave Software, offers a new feature, which allows developers to prioritize, review, and triage issues in their projects. Based on a sophisticated analysis of the code, including factors such as analysis complexity, Klocwork 2017 can identify and apply a recommendation to detected issues so developers can be confident that the identified issues are true and should be tackled first. And because Klocwork 2017 uses many factors to calculate the triage suggestions, whether it's a desktop analysis or a continuous integration build, the triage can change for individual issues depending on how the code has changed.

Because developing for the embedded market is so complex, and getting more so every day, testing will always be a large, important task. Tools like Klocwork 2017 will continue to make this process less of a burden so developers can cut through the noise, fix what really matters, and move on to developing the next important part of a project.



USA **800-487-3217**
FRANCE **+33 (0) 1 46 93 94 20**
GERMANY **+49 (0) 6103 5934 0**
UK **+44 (0) 8450 549950**
JAPAN **+81 (0) 3 5211 7760**

www.roguewave.com

© 2017 Rogue Wave Software, Inc.
All Rights Reserved

Rogue Wave Software is the largest independent provider of cross-platform software development tools, components, and platforms in the world. Through decades of solving the most complex problems across financial services, telecommunications, healthcare, government, academia, and other industries, Rogue Wave tools, components, platforms, and services enable developers to write better code, faster.