RogueWave SOFTWARE
Accelerating Great Code

Elektrobit

CASE STUDY: ELEKTROBIT

From false positives to trust: A story of SCA adoption

INDUSTRY
Key supplier of automotive components and software

APPLICATION
Large-scale C, C++, and C# projects

WEBSITE
www.elektrobit.com

CHALLENGES
Deliver embedded systems with high safety and security

Improve false positive rates for static code analysis

Get programmers to buy into static code analysis

SOLUTION
Klocwork

RESULTS
Low false positive rates even for large code bases

Improved utilization of static code analysis across teams

At the heart of the Internet of Things (IoT) are the embedded devices that feed the systems, connections, and interactions the world depends on. Elektrobit is at the heart of embedded, supplying HMI technologies, navigation, electronic control units (ECU), and software for over 70 million vehicles and over 1 billion embedded devices. With increasing customer demands for openness, connectivity, and integration, topics such as safety, security, and vulnerability defense have moved into the spotlight for the whole chain of suppliers and OEMs in the automotive industry. This massive footprint and complexity demands high safety, security, and reliability standards, which means an effective analysis solution must be used as early as possible within the software development lifecycle (SDLC) to avoid costly recalls for products in the field. When Elektrobit chose to adopt a new static code analysis (SCA) tool, it involved a mix of technical evaluation and programmer education to get the right tool and the right people onboard within the organization.

Why use SCA?

Elektrobit touches nearly every car on the road today, whether it's an assistance system invisible to the driver or navigation technology built for complex human interaction. Over 25 years, Elektrobit has delivered products that major car manufacturers and suppliers trust to perform and perform safely. The cost of quality or functional safety defects in the field for systems that are hard to update is very high, with the potential for damaging trusted relationships with customers. With growing systems complexity and accelerating release cycles, Elektrobit's software engineering group needed a SCA solution that prevented code defects before they reached testing or the customer.

Enter static code analysis. Alexander Much, head of software systems engineering at Elektrobit used his experience as a former programmer to bring SCA to the process improvement table. Code improvement and safety was foremost on his mind but he knew that new tools would be a hard sell to programmers. "Defects and recalls are expensive and we can't afford to have them. My experience has taught me that if you're not using SCA nowadays, you're not state-of-the-art. This isn't the first time we've introduced SCA at Elektrobit and the most important thing is gaining trust of the programming teams." Like Elektrobit's products, adopting SCA in the development process was as much about the technology as it was about establishing the confidence of programmers.

"Defects and recalls are expensive and we can't afford to have them. My experience has taught me that if you're not using SCA nowadays, you're not state-of-the-art. This isn't the first time we've introduced SCA at Elektrobit and the most important thing is gaining trust of the programming teams."

Alexander Much, head of software systems engineering at Elektrobit

The measure of a tool

With strict software compliance requirements from automotive OEMs and a challenging code base, Elektrobit knew they had to select the right SCA tool that met multiple needs. For example, programming teams had been using less sophisticated tools to identify defects before but the number of reported issues and the lack of tool support in visualization and explanation made it difficult to distinguish between a real defect and a false positive. They also have a very large code base and multiple inputs from internal development teams, suppliers, and even open source code. Mr. Much explains, "Automotive OEMs have a clear idea of what is needed and we have to push those needs on suppliers. In the end, it's our responsibility to make sure they comply with all the standards. The SCA tool simply has to handle that."

A rigorous tools evaluation process was undertaken to measure performance against different aspects of this challenging environment: a very large code base to start with, integrating large amounts of supplied code, and avoiding past experience with large numbers of false positives, among others. Taking an incremental approach, the evaluation team first ran a set of rules selected by the software architects and, once results were acceptable, moved on to the next set. After a three-month evaluation, Klocwork was selected as the tool of choice. "On 2 to 3 million lines of code, we ran Klocwork against multiple competitors. It handled the code base very well and had the ability to deal with C++ templates in a useful way," says Mr. Much. "In comparison, there was also a low rate of false positives. At some point during tool evaluation, the price discussion becomes relevant and you need to take the workload of the programming teams using a SCA tool into account. A cheaper tool with fewer capabilities and a higher number of false positives can generate a significant amount of unnecessary work in the programming and quality assurance teams. This easily outweighs the price difference. In that case, it's definitively worth spending more money up front. After all you don't give a lumberjack a blunt axe and expect the same amount of trees cut."

Other benefits came into play as the adoption of Klocwork grew, owing to the complexities of the environment that are unique to every development team. "Visual Studio had to be used for a customer and the tool integration process was slow. Rogue Wave and our integration partner Emenda were on site as much as possible to make sure things were corrected quickly, working with each programming team until it was acceptable. We also worked directly with the product managers on critical issues or new features, which helped build the roadmap of Klocwork."

This relationship was an important piece of the puzzle, as static code analysis was a critical component in delivering robust products. As Mr. Much says, "An SCA tool is not a simple tool you buy, you get a little married to it. It can only be successful if it has users that provide feedback on how it's being used and provide ideas for improvement. The tool gets better with the amount of analyzed code bases and vice versa. Our relationship with Klocwork has been successful because we have that trust on all sides."

"An SCA tool is not a simple tool you buy, you get a little married to it. It can only be successful if it has users that provide feedback on how it's being used and provide ideas for improvement. The tool gets better with the amount of analyzed code bases and vice versa. Our relationship with Klocwork has been successful because we have that trust on all sides."

Alexander Much, head of software systems engineering at Elektrobit

Side effects

An unforeseen insight from using Klocwork was how SCA could improve the software architecture. Mr. Much explains, "The interplay between local and global is very important. Klocwork has a more global view on the source code rather than on the local environment, for example, a line, function, or a class. This can reveal shortcomings in the overall software architecture and provide information that is much more valuable to the overall product. If I could go back and change something, I would suggest getting the global view in the software architecture in a better shape to enable SCA tools to provide more precise results on the local level."

Up and down the supplier chain, Elektrobit values trust and confidence in tools, just as their customers expect it of them, and Klocwork was a natural fit. Mr. Much has the final word, "Working with Emenda and Rogue Wave is a great relationship. Klocwork is stable in our environment and is accepted by the programming teams. It's a success story, as simple as that."

RogueWave SOFTWARE

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

Rogue Wave provides software development tools for mission-critical applications. Our trusted solutions address the growing complexity of building great software and accelerates the value gained from code across the enterprise. The Rogue Wave portfolio of complementary, cross-platform tools helps developers quickly build applications for strategic software initiatives. With Rogue Wave, customers improve software quality and ensure code integrity, while shortening development cycle times.

About Emenda, Ltd.
Based throughout the heart of Europe, Emenda specializes in the provision of software engineering tools and consultancy services for the development and verification of high integrity software systems to customers worldwide. Learn more at www.emenda.eu.