

AUTOSAR C++ COMPLIANCE MODULE

THE AUTOSAR COMPLIANCE MODULE EXTENDS QA·C++ FOR OUT-OF-THE-BOX APPLICATION OF THE AUTOSAR CODING GUIDELINES.

As a partnership of over 180 automotive manufacturers, automotive suppliers, tool vendors and semiconductor vendors, AUTOSAR (AUTomotive Open System ARchitecture) aims to standardize and future-proof basic software elements, interfaces and bus systems, to help vehicle manufacturers manage growing system complexity while keeping costs down. It develops standardized open software architectures for automotive Electronic Control Units (ECUs).

PRQA is AUTOSAR's static analysis development partner. We contributed our expertise in the C++ programming language and best-practice software development gained over the last 30 years to the development of AUTOSAR's "Guidelines for the use of the C++14 language in critical and safety-related systems". Following the initial draft of these coding guidelines, AUTOSAR used PRQA's QA·C++ static analyzer to ensure the quality of the demonstration code for its 'Adaptive Platform' standard for connected and autonomous vehicles. The valuable insights gained from this activity, combined with our unique contribution to the coding guidelines have enabled the development of the only static analysis solution that has been optimized for AUTOSAR-compliant software development.

SOLUTION OVERVIEW

QA·VERIFY

A complete code quality management control center, QA·Verify guarantees that all team members consistently apply the coding guidelines in addition to tracking and reporting code quality for the duration of your project.

QA·C++

The most advanced static analyzer available for modern C++, QA·C++ can ensure your code complies with the AUTOSAR coding guidelines as well as a broad range of other commonly adopted standards. In addition to applying coding standards, QA·C++ performs the deepest, most precise analysis of your program's flow. It catches bugs that go undetected by manual reviews and other tools, while generating the lowest possible number of false positive diagnostics.



AUTOSAR C++ COMPLIANCE MODULE

The AUTOSAR Compliance Module extends QA·C++ for out-of-the-box application of the AUTOSAR coding guidelines. The guidelines permit some of the language features prohibited by some earlier standards. Examples include: Dynamic memory, exceptions, templates, inheritance and virtual functions. There are rules to ensure that these language features are used only in a safe manner.



QA Systems and Programming Research Ltd

QA Systems is an authorised reseller of the QA·C / QA·C++, QA·Verify static testing tools and their compliance module add-ons, which are owned by Programming Research Ltd.

QA·C®, QA·C++® and QA·Verify® are registered trademarks of Programming Research Ltd. These tools and this document are the copyright © 2016 of Programming Research Ltd.

Third party trademarks, logos and trade names appearing in this document are the trademarks and property of their respective owners.

QA·C, QA·C++ and QA·Verify, offer the closest possible examination of C and C++ code. All contain powerful, proprietary parsing engines combined with deep accurate dataflow which deliver high fidelity language analysis and comprehension. They identify problems caused by language usage that is dangerous, overly complex, non-portable or difficult to maintain. Plus, they provide a mechanism for coding standard enforcement.

Contact Us

For further information regarding QA·C, QA·C++ and QA·Verify and compliance module add-ons, please contact QA Systems at info@qa-systems.de.