Qt Quality Assurance

Axivion Suite

Axivion Static Code Analysis
Axivion Architecture Verification

A short overview outlining the features and benefits of our outstanding static code analysis tools.

Axivion Suite

▎⦙⦙⦙ Axivion Static Code Analysis

▨▨ Axivion Architecture Verification

Your benefits

Enhance the efficiency and productivity of your team and maintain the high quality of your code.

✓ Automated quality assurance which reduces defects, code footprint and dependency on individuals

✓ Holistic and comprehensive analysis for increased productivity and ROI by saving time, money and resources

✓ Fully customisable and easy to integrate into your existing development environments

✓ Scalable tool, which adapts to changing requirements and grows with your needs

✓ Follows industry-leading safety & security standards to ensure your code is compliant

✓ Targeted review with detailed delta-analysis as part of the CI process e.g. before code is committed or to monitor improvements and prevent software erosion

✓ Helps ensuring cyber security by checking code compliance with CERT, CWE and other security guidelines

✓ On-time delivery of feature releases of your software thanks to risk mitigation

✓ Unmatched service and support to ensure you benefit right from the start, e.g. support during roll-out from our experienced Service Team

Software Erosion

Software erosion (also known as technical debt) is a process through which maintainability, extensibility and reusability of software is made more difficult or even impossible over time. Source code is continuously modified to enhance the functionality of the software to adapt to new or changed requirements. Although this might not impair the function of the software, it potentially introduces additional burden on future tasks.

As a consequence, software erosion can also degrade the observable operation of a system and compromises its functional safety and security. The result: relevant criteria, e.g. for ISO, DOD, MISRA, may not be met anymore, thus preventing successful certification in certain application domains.

Axivion Static Code Analysis

The automated analysis of your software projects identifies violations of coding guidelines according to MISRA (including MISRA C: 2023 and MISRA C++:2023) and AUTOSAR C++14. Security-relevant violations are analysed with coding guidelines according to CERT®, C Secure Coding and CWE. Metric violations are displayed and documented, as are clones, cycles or unreachable code.

SGS-TÜV Saar GmbH has certified that Axivion Static Code Analysis is suitable for use in the development of safety systems up to the highest level of the safety requirement contained in the respective standard: ISO 26262 up to ASIL D, IEC 61508 up to SIL 4, IEC 62304 up to Class C and EN 50657 up to SIL 4.
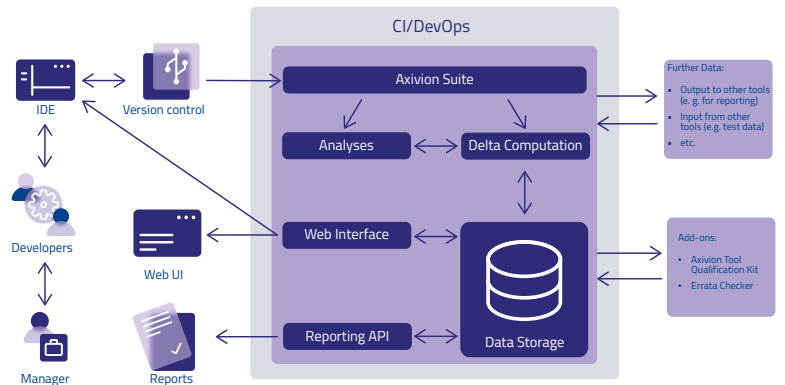
Integration into your existing development environment and processes ensures success through fast and focused feedback
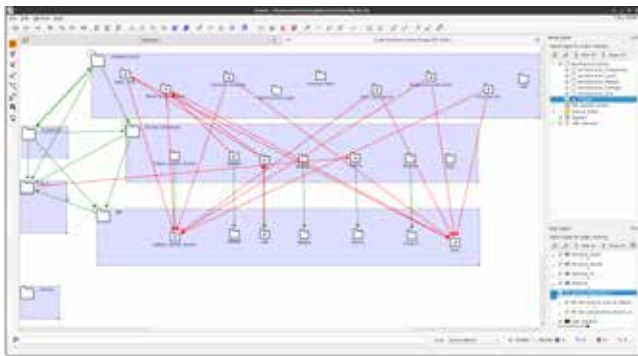
Validate or even reconstruct your architecture

Axivion Architecture Verification

Software architecture and design need to match with the code for you to be sure you can use the software architecture as a guide and baseline for discussing the impact of new features. Only then is a long-term targeted and planned development of your products possible.

Axivion Architecture Verification ensures your code complies with your architecture. In addition to the functional architecture, the tool also reviews and checks safety and security architecture specifications for compliance, e.g. Freedom from Interference.

More than the sum of its parts…

Axivion Suite is the combination of Axivion Static Code Analysis and Axivion Architecture Verification and will provide you with an industry-leading, in-depth analysis of your code.

Add to this the Axivion Tool Qualification Kit to enable the specific check and verify the suitability in environments with functional safety requirements.

Seeing is believing….

Let us show you how Axivion Suite will protect your code from software erosion. Based on your specific requirements our experts can evaluate your ROI when using Axivion Suite.

Request a meeting via our website: www.qt.io/axivion
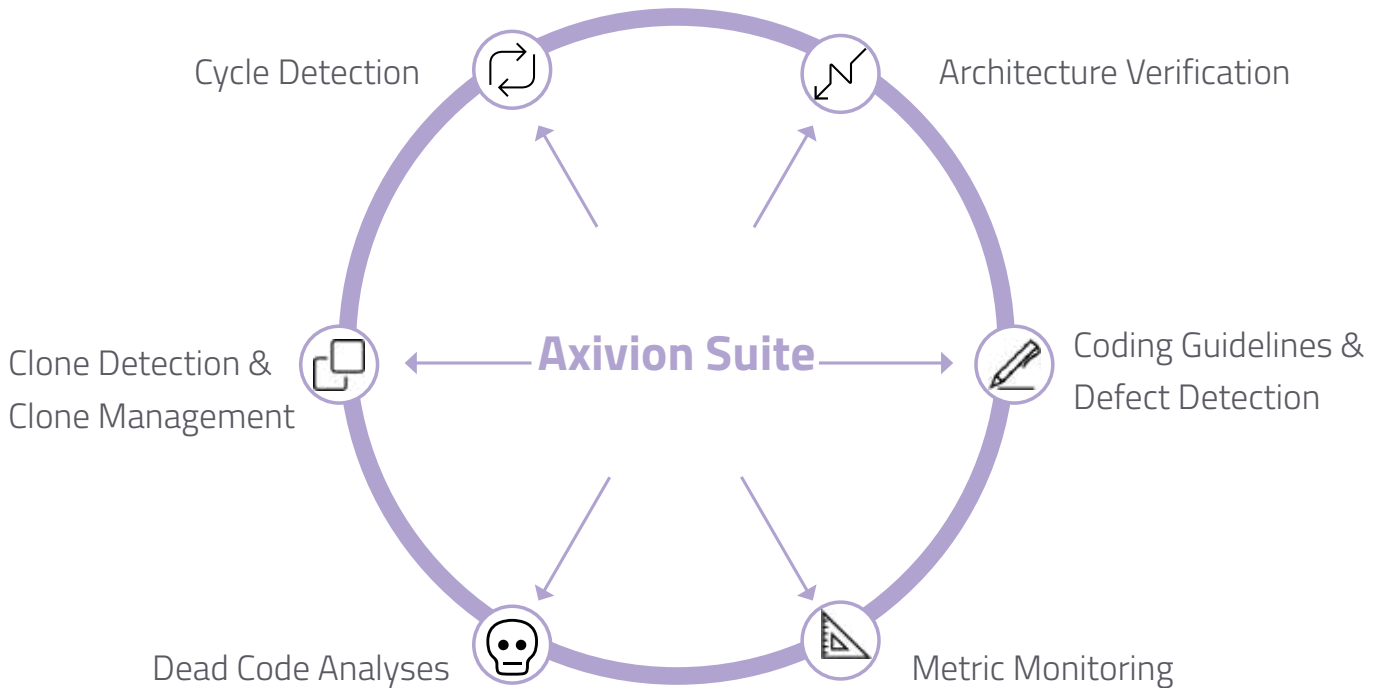
Qt Quality Assurance

Stopping software erosion

Future-proof your software development.

Axivion Suite checks software projects for style and coding violations. Detecting clones, dead code, division by zero and other defects in the code at an early stage drastically reduces the time and costs involved fixing these issues at a later stage.

Axivion Suite also supports a vast number of metrics and coding guidelines. Especially safety-relevant software can therefore be easily monitored. Furthermore, individual rules, configuration options and coding guidelines can be added, ensuring high quality standards are maintained right from the start.

New violations can easily be identified thanks to the delta-analysis. This way, baselining is made easy. Violations are displayed on the user-friendly dashboard. These results can also include findings from 3rd-party tools.

Cycle Detection

Architecture Verification

Clone Detection & Clone Management

Axivion Suite

Coding Guidelines & Defect Detection

Dead Code Analyses

Metric Monitoring

Scalable static code analysis with abstract interpretation and pointer analysis

Key features

Architecture verification

- Integrated Modeler
- Interfaces to UML Tools
- AUTOSAR XML Import (ARXML)
- Freedom from Interference
- Architecture Reconstruction
- Architecture views for safety and security

Coding guidelines

- MISRA C/C++
- AUTOSAR C++14
- SEI CERT® C/C++
- C Secure Coding
- CWE
- Qt-specific rule sets
- Best Practice (e.g. CQM)
- Customised Checks

Metrics monitoring

- HIS
- OO Design Metrics
- Complexity Metrics
- Best practice and customised checks

Dead code analysis

- Reachability Analysis
- Analysis of Libraries

Clone detection & management

- Type I (1:1)
- Type II (Parametrizations)
- Type III (Parametrizations + Add/Delete)

Cycle detection

- Calls
- Module dependencies
- Includes/Imports
- Customised checks

Defect detection

Memory and Pointers

- NULL dereferences
- Escaping addresses of local variables
- Memory leaks from new/malloc without delete/free
- Mismatched resource allocations/releases
- Resources used for reading/writing at the same time
- Forbidden operations on resources
- Double free/use after free
- Comparison/subtraction of unrelated pointers
- Array access out of bounds
- Buffer overflow
- Side effects
- Taint analysis
- Race condition analysis

Exceptions

- Exceptions during stack unwinding
- Violations of exception specifications
- Uncaught exceptions
- Dead catch blocks

Numerical errors

- Divisions by zero
- Overflow in arithmetic computation
- Assignment of bad values to enum-typed variables

Logical errors and customised rules

- Forbidden argument values
- Uninitialised variables
- Unused definitions
- Constant conditions

For the Axivion tools "defects" focus on run time errors. But the tools, of course, also cover API usage, syntax errors etc.

Please contact us for further details.

Qt Quality Assurance

Safety & Security

Do you need to show that your code is suitable for use in safety-critical environments?

With Axivion Suite you can.

Axivion Tool Qualification Kit

The Axivion Tool Qualification Kit helps to achieve the tool confidence level (TCL) required by the respective safety regulation (e.g. ISO 26262). By automatically carrying out validation test it ensures that new code always meets the set safety standards.

Axivion Suite is suited for use in the development of safety systems. Axivion Static Code Analysis (SCA) is certified by SGS-TÜV Saar GmbH for the following safety-relevant standards:

- ISO 26262 (up to ASIL-D)
- IEC 61508 (up to SIL 4)
- EN 50128 (up to SIL 4)
- EN 50657 (up to SIL 4)
- IEC 62304 (up to Class C)

More than a tool

Unmatched service to ensure you benefit right from the start.

Axivion Suite is highly customisable and can be configured to match your needs. To get the most out of it, it requires an optimised adoption for each individual environment. The roll-out is therefore always accompanied by our services team (remotely or on-site). Additionally, we offer the following services:

- Customisation of coding guidelines
- Implementation of project-related coding guidelines
- Training of employees on how to use Axivion Suite
- Basic training courses, e.g. software architecture, coding guidelines
- Support for architecture verification, tool classification and qualification (e.g. ISO regulations) etc.

Our service team works closely with the in-house R&D department to support customers and develop customised solutions.

Made for you

Axivion Suite is used in a wide range of industries.

Whether you are developing embedded software for small sensors or large machinery, aiming for improved usability or want to meet industry standards: Axivion will help you achieve your goals faster, better and with fewer resources.

- Automotive
- Aviation, Aerospace & Defence
- Construction
- Electronics
- Energy technology
- Industrial Applications
- IT, telecommunications & software
- Measurement & control technology

- Mechanical engineering
- Medical technology
- Pharmaceutical & lab technology
- Railway technology
- Research institutions

Basic technical specifications

Note: This is just a rough overview and refers to Axivion Suite 7.7

Please contact us for a complete list of specifications.

Supported languages and compilers

Languages	C, C++, C#[1]
Compilers	Blackfin, Clang, Codevision, CodeWarrior®, Cosmic, Green Hills Software®, GNU, IAR™, Keil™, Microchip®, Microsoft®, Renesas, Tasking, TI, Windriver, Others

Supported operating systems

Host OS	Windows® 7/8/10/11, Windows® Server® 2008 R2/2012/2016/2019/2022 x86_64 GNU/Linux® (minimum requirement is glibc2.24 or later) macOS® (minimum requirement is macOS 10.15)

Plugins

IDE	CLion, Eclipse™, Eclipse-based (e.g. Atollic TrueSTUDIO®, CodeWarrior®, DAVE™, STM32CubeIDE, TI Code Composer Studio™), Microsoft® Visual Studio®, Microsoft® Visual Studio Code®, Generic plugins
CI/DevOps	Azure® DevOps, Jenkins®, Integration for e.g. Bitbucket®, GitLab®, GitHub

Supported version control systems

Version Control System	Borland®/Inprise®/MicroFocus® StarTeam®, CVS, Fossil, Git™, IBM® Rational® ClearCase®, IBM® Rational®, Team Concert®, Mercurial, Microsoft® Team Foundation Server®, Microsoft® Visual SourceSafe®, MKS Source Integrity®, Perforce®, Perforce®/Seapine® Surround®, Plastic, PTC Integrity®, Serena® Dimensions®, Serena®, PVCS®, Subversion®

Supported UML® tools

UML® Tools	IBM Rational Rhapsody, Sparx Enterprise Architect (via XMI or .qea-files)

Supported standards and guidelines

Coding Guidelines	MISRA C:2004/2012/2019/2023, MISRA C++:2008/2023, AUTOSAR C++14, CERT C Rules, CERT C Recommendations, CERT C++ Rules, CWE, ISO / IEC TS 17961, Qt Framework, Customised checks and other
Quality Guidelines[2]	ISO 26262 up to ASIL-D IEC 61508 up to SIL-4 IEC 62304 up to Class-C EN 50128, EN 50628

Other

Supported browsers	Microsoft® Edge, Mozilla Firefox®, Google Chrome™
Requirements	Python (3.8.1 - 3.12) Java® Runtime (8, 11-14 and 17)
Add-on	Axivion Tool Qualification Kit

[1] C# support is limited
[2] Certified by SGS-TÜV Saar

Qt Group (Nasdaq Helsinki: QTCOM) is a global software company, trusted by industry leaders and over 1.5 million developers worldwide to create applications and smart devices that users love. We help our customers to increase productivity through the entire product development lifecycle: from UI design and software development to quality management and deployment. Our customers are in more than 70 different industries in over 180 countries. Qt Group employs some 700 people, and its net sales in 2022 were 155.3 MEUR. To learn more, visit www.qt.io.

www.qt.io/axivion