

Axivion for CUDA

Unlock the Full Potential of Your CUDA Applications



Automated Static Code Analysis and Architecture Verification for Your CUDA Applications

CUDA has become a cornerstone of today's high-performance computing, powering applications across various industries. As these applications scale, raw performance is no longer enough. **Safety, security, reliability, and maintainability** are now essential for sustainable development.

Whether you need to maintain high software quality, ensure software segregation and freedom from interference or need to comply with guidelines and regulations: **Axivion always includes your CUDA C++ code** during the analysis.

Minimize Technical Debt

Axivion for CUDA helps you reduce existing technical debt, and **prevents new debt from developing**. It integrates seamlessly into your current development environment and offers the flexibility to adapt and scale with your evolving needs.

Most static analysis tools treat CUDA as a generic C/C++ extension, missing its unique constructs and challenges. Axivion for CUDA is **designed specifically for GPU applications**, offering full support for CUDA syntax, architecture verification, and compliance with industry standards.

It **integrates seamlessly into CI/DevOps workflows**, helping teams maintain clean, compliant code without sacrificing speed or innovation.



Ensure Compliance

Comply with **NVIDIA's CUDA C++ guidelines** as well as other leading guidelines and industry standards.



Enhance Maintainability

Reduce time needed to fix **technical debt** by 33% and keep your code maintainable.



Maximize Efficiency

Save up to 15% in **development costs** by making sure your software architecture is easy to understand.



Detect Bugs Early

Save time and money by detecting bugs early. Fix them before they cause damage.



Reduce Your Team's Workload

Let your developers **focus on building features** instead of fixing bugs.



Faster Time to Market

Surpass your competitors, by going to market faster with a higher-quality product.



Contact us

Do you have questions?

Are you interested in a demo?

Axivion for CUDA

Purpose-Built for GPU Applications

Ideal for Safety-Critical Programming

Axivion ensures **compliance with standards and guidelines** such as MISRA, ISO 26262, IEC62304, CERT, CWE and, of course, NVIDIA's CUDA C++ Guidelines. Even without detailed knowledge about the rules, it is easy to meet the requirements and develop safe and secure software.

Automated **multi-language support** enables complete analysis across mixed-language codebases, thereby **eliminating the need for manual reviews** or applying multiple tools.

The Axivion **Tool Qualification Kit** supports automated validation and requalification after updates. It helps qualify the entire toolchain, covering both static code analysis and architecture verification for CUDA-based projects.

Your All-In-One Tool for C, C++ and CUDA C++

- ✓ Architecture Verification
- ✓ Clone Detection and Management
- ✓ Metric Monitoring
- ✓ Coding Guidelines & Defect Detection
- ✓ Dead Code Analysis
- ✓ Cycle Detection

Architecture Verification

Keep your software architecture clear and maintainable by **automatically detecting deviations from the intended structure** after each code change. Existing architecture debt can be addressed through targeted refactoring, with progress tracked in detailed result reports.

Static Code Analysis

Axivion Static Code Analysis is ideal for developing safety-critical software. It automates repetitive, full-scale checks and **delivers reliable results through comprehensive reports**. By detecting issues early, Axivion helps prevent technical debt, conserve resources, and maintain high code quality.

Key Technical Specifications

Languages	CUDA® C++, C, C++
Compilers	Blackfin, Clang, Codevision, CodeWarrior®, Cosmic, Green Hills Software®, GNU, IAR™, Keil™, Microchip®, Microsoft®, NVIDIA nvcc, Renesas, Tasking, TI, Windriver, Others
Host OS	Windows® / Windows® Server®, GNU/Linux®/Linux® ARM, macOS®/macOS® ARM
Browsers	Microsoft® Edge, Mozilla Firefox®, Google Chrome™
IDE plugins	Qt Creator, CLion, Eclipse™, Eclipse-based (e.g. e² Studio, Atollic TrueSTUDIO®, CodeWarrior®, DAVE™, STM32CubeIDE, TI Code Composer Studio™), Microsoft® Visual Studio®, Microsoft® Visual Studio Code®, Generic plugins
CI/DevOps plugins	Azure® DevOps, Jenkins®, Integration for e.g. Bitbucket®, GitLab®, GitHub
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®
UML® Tools	IBM Rational Rhapsody, Sparx Enterprise Architect (via XMI or .qea-files), PlantUML
Coding Guidelines	NVIDIA's CUDA C++ Guidelines for Robust and Safety-Critical Programming, MISRA C/MISRA C++, AUTOSAR C++14, CERT C/CERT C++, CWE, ISO / IEC TS 17961, Qt Framework, Custom rules
Quality Guidelines	ISO 26262, IEC 61508, IEC 62304, EN 50128, EN 50628

Note: The above specifications apply to Axivion 7.11 and are not exhaustive. For a complete list of specifications, please contact us. Technical data is subject to change without prior notice. All rights reserved. All company and/or product names are trademarks and/or registered trademarks of their respective manufacturers in their markets and/or countries. We are constantly making efforts to deliver the latest status of data to our partners. Specifications may change in the time between the product release and the release of this document. © Qt Group | 202511