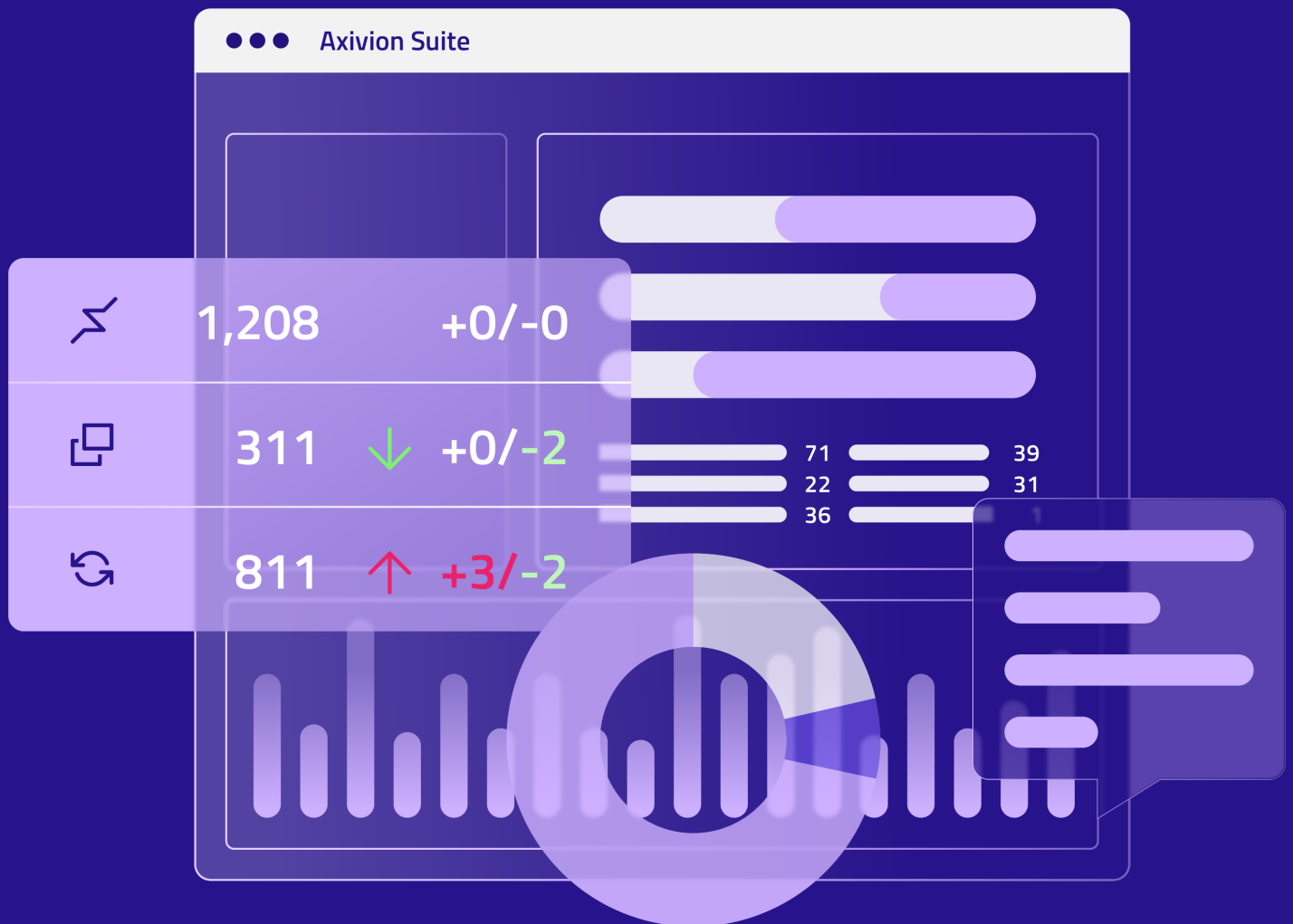


# Axivion Suite

⌘ Axivion Static Code Analysis

⌘ Axivion Architecture Verification

静的コード解析とアーキテクチャ検証ツールの特徴と利点



# Axivion Suite

- ☑ 詳細な差分解析を備えた、包括的な静的コード解析とアーキテクチャ検証を実現する単一ツール
- ☑ 生産性と投資対効果 (ROI) を高める、自動化されたソフトウェア品質チェック
- ☑ MISRAなどの業界をリードする標準やガイドラインに準拠し、ソフトウェアのコンプライアンスを確保
- ☑ 完全にカスタマイズ可能で、既存の開発環境へ容易に統合可能
- ☑ 開発者の日常業務を強化する、AI 活用ワークフローに対応
- ☑ 変化する要件に適応し、開発ニーズとともに進化するスケーラブルなツール
- ☑ セキュア・バイ・デザインの原則に基づき、セキュリティ関連の脆弱性を特定することでCRAコンプライアンス対応を支援
- ☑ 導入初日から高い生産性を実現する、比類のないサービスとサポート

## ソフトウェアの劣化とは？

ソフトウェアの劣化(技術的負債とも呼ばれる)とは、ソフトウェアの保守性、拡張性、再利用性が、時間の経過とともに困難になる、あるいは不可能になるプロセスのことです。ソースコードは、新しい要件や変更された要件に適応するために、ソフトウェアの機能を強化するために継続的に修正されます。これによってソフトウェアの機能が損なわれることはないかもしれませんが、将来のタスクにさらなる負担を強いる可能性があります。

ソフトウェアの劣化は、システムの動作をいちじるしく劣化させ、機能安全やセキュリティを損なう可能性もあります。結果として、ISO、DO-330、MISRAなどの関連基準が満たされなくなり、特定のアプリケーション領域で認証が取得できなくなる可能性があります。



### バグの早期検出

バグを早期に検出し、重大な問題になる前に修正することで**時間とコストを節約**



### 効率の最大化

ソフトウェアアーキテクチャを理解しやすくすることで、**開発コスト**を最大15%削減



### コンプライアンスの確保

**NVIDIAのCUDA C++ガイドライン**に加え、その他の主要なガイドラインや業界標準に準拠



### チームの負荷軽減

開発者がバグ修正ではなく**機能開発に集中**



### 市場投入までの時間短縮

より高品質な製品をより早く市場に投入し、**競合優位**を獲得



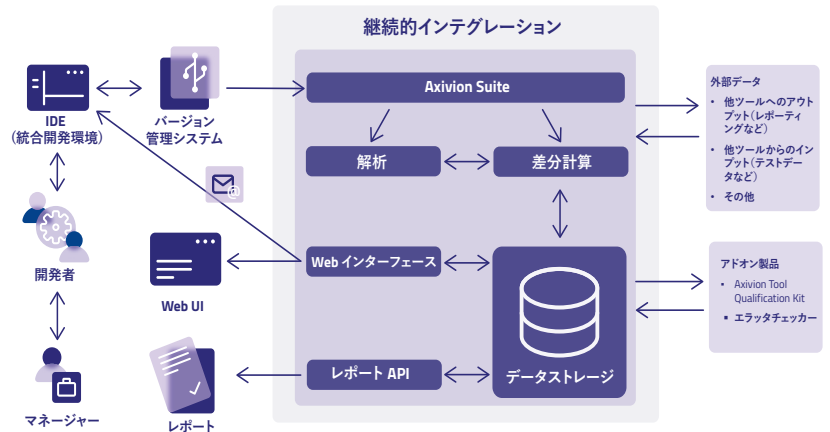
### 保守性の向上

**技術的負債**の修正に必要な時間を33%削減し、コードの保守性を維持

## ニーズに合わせた、柔軟なルール設計

Axivion Suite は、Axivion 静的コード解析 と Axivion アーキテクチャ検証を組み合わせたもので、業界最先端の詳細なコード解析を提供します。既存の開発環境にシームレスに統合でき、高いカスタマイズ性を備えているため、企業独自のルールや追加のルールセットを、すでに非常に充実したコーディングガイドラインに組み込むことが可能です。

さらに、当社の専門家がロールアウトプロセス全体をサポートし、初日からその価値とメリットを最大限に活用していただけます。



AIを活用したワークフローに Axivionを組み込みたい場合は、Model Context Protocol (MCP) コネクタを有効にして、お好みの大規模言語モデル(LLM)をご使用ください。AIエージェントがAxivionのドキュメントや解析結果にアクセスできるようになり、開発者がバグをより迅速に理解・修正できるようサポートします。エージェントをその他のソース(インターネット、社内サーバーなど)に接続するかどうかは、お客様が制御できます。

VS Code用プラグインを使用することで、開発者はコードを理解するためにAIを適用するだけでなく、AIの支援を受けてコードを変更することもできます。

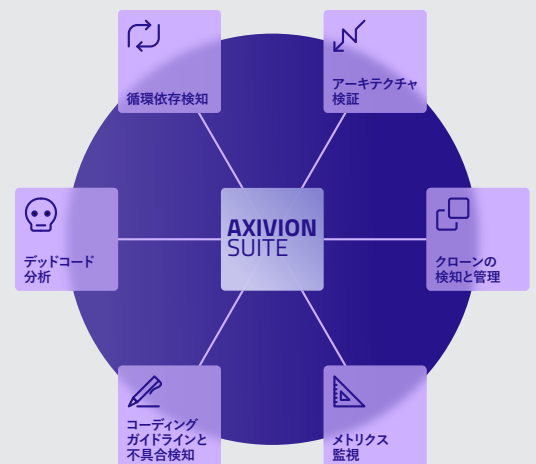
**AIで強化されているが、AI依存ではない:** Axivionは解析にAIを使用していないため、安全性が重要な環境でのソフトウェア開発およびコンプライアンス検証に適しています。

## ソフトウェアの劣化を阻止

Axivion Suite は、ソフトウェアプロジェクトのスタイル違反やコーディング違反をチェックします。クローン、デッドコード、ゼロ除算、その他のコードの不具合を早い段階で検出することで、後の段階でこれらの問題を修正する時間とコストを大幅に削減します。

Axivion Suite は、膨大な数のメトリクスとコーディングガイドラインもサポートしています。そのため、特に安全性に関連するソフトウェアを容易に監視できます。さらに、個別のルール、設定オプション、コーディングガイドラインを追加できるため、最初から高い品質基準が維持されます。

差分解析により、新たな違反を簡単に特定することができます。このようにして、ベースライン化が容易になります。違反は、ユーザーフレンドリーなダッシュボードに表示されます。これらの結果には、サードパーティツールからのデータも含めることができます。



## 組み合わせによる相乗効果

### Axivion 静的コード解析

#### 安全なソフトウェアのための高品質なコード

Axivion静的コード解析は、ソフトウェアプロジェクトの自動解析により、MISRA (MISRA C:2025とMISRA C++:2023を含む)、CUDA C++ガイドライン、およびAUTOSAR C++14 に準拠したコーディングガイドラインの違反を特定します。

セキュリティ関連の違反は、CERT®、Cセキュアコーディング、CWEに準拠したコーディングガイドラインで分析されます。メトリック違反は、クローン、サイクル、到達不能コードと同様に表示され、文書化されます。

#### セーフティクリティカルなソフトウェア開発に最適



SGS-TÜV Saar GmbH は、Axivion静的コード解析が、各規格に含まれる最高レベルの安全要件に対応した安全システムの開発に適していることを認定しました：

- EN 50716 (SIL 4 まで)
- IEC 61508 (SIL 4 まで)
- IEC 62304 (Class C まで)
- ISO 26262 (ASIL-D まで)



認証書をダウンロード:

### Axivion アーキテクチャ検証

新機能の影響を検討するためのガイドおよびベースラインとしてソフトウェア・アーキテクチャを使用するためには、ソフトウェア・アーキテクチャとデザインが、コードと一致している必要があります。そうすることで、初めて長期的な目標を持った計画的な製品開発が可能になります。

#### アーキテクチャチェック

Axivionアーキテクチャ検証を利用することで、コードがアーキテクチャに準拠していることを確認できます。

機能的なアーキテクチャに加えて、安全性とセキュリティのアーキテクチャ仕様もレビューし、FFI (Freedom from Interference) などのコンプライアンスをチェックします。

#### アーキテクチャの再構築

ドキュメントが存在しない場合でも、アーキテクチャを復元できます。レガシーコードやサードパーティコードを扱う際に特に役立ちます。

#### アーキテクチャのエクスポート

コードから必要なすべての要素(コンポーネント、関数、情報フロー、依存関係など)を抽出し、Enterprise Architectで開いて編集できます。

## ニーズに応じたライセンス

Axivion Suiteは、中規模企業から大企業まで、さまざまなニーズに応える、幅広い業界で信頼されています。

小型センサー用の組み込みソフトウェアから大型機械までの開発、使いやすさの向上を目指す場合や、ガイドラインや規格の遵守を求める場合でも、Axivionはニーズに合わせたツールで、より速く、より良く、より少ないリソースで目標達成をサポートします。



無料デモを依頼:

### Axivion アーキテクチャ検証

強固な基盤を構築し、コードがアーキテクチャに適合していることを確認

#### 静的コード解析

デッドコード解析、コーディングガイドライン、メトリクス監視

#### 静的コード解析 PRO

デッドコード解析、コーディングガイドライン、メトリクス監視、循環依存検出、クローン検出および管理

#### Axivion Suite

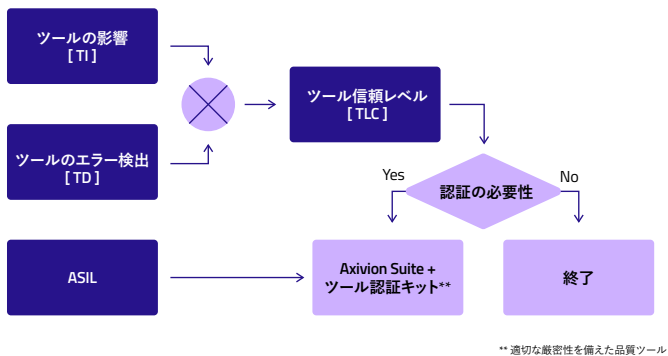
上記すべての機能

#### Axivion for CUDA

CUDAプロジェクトにおけるソフトウェアの劣化を防止し、NVIDIAのCUDA C++ ガイドラインに準拠するための Axivion Suite

## Axivion ツール認定キット

Axivion ツール認定キットは、各安全規制で要求されるツール信頼性レベル (TCL) の達成を支援します。検証テストを自動的に実施することで、新しいコードが常に設定された安全基準を確実に満たすことができますようにします。



ツール認定キットは、ツールチェーンの分類および認定を支援します。本キットには、HIS メトリクス、MISRA、AUTOSAR C++14、CUDA C++ ガイドライン、CWE、C セキュアコーディング、CERT ルール向けの事前構成済みテストスイートが含まれており、テストの実行および結果評価プロセスを自動化することが可能です。

また、アーキテクチャ検証をチェックおよび妥当性確認するためのテストも含まれています。

Axivion ツール認定キットは、機能安全要件を持つ環境において、Axivion 静的コード解析、Axivion アーキテクチャ検証、ならびに CUDA 向け Axivion の適合性を検証することができます。

### 静的コード解析向けに提供されるテストファイル

- AUTOSAR
- CERT C/C++
- CWE
- エラーチェック
- HIS メトリクス
- MISRA-C:2012/2019/2023/2025
- MISRA-C++:2008/2023
- NVIDIA CUDA C++ ガイドライン
  - CUDA コードに対して MISRA-C++:2023 ルールを適用するためのテストファイルを含む
- セキュアコーディング

### アーキテクチャ検証向けに提供されるテストファイル

- Architecture\_Analyses
  - アーキテクチャ解析、サニティチェック、モデル変換
- Dependencies
  - 型情報およびアクセス情報を付加することで RFG (Resource Flow Graph) を拡張
- Exporters
  - GXL 形式でのエクスポートおよび RFG の保存
- Graph\_Based\_Analyses
  - サイクル検出、デッドコード検出、RFG スキーマチェックなど、RFG ベースの追加解析
- Importers
  - Enterprise Architect や PlantUML など、さまざまなソースを対象としたインポーター
- Legacy
  - レガシールール
- Transformations
  - グラフ変換

## CRA 対応への取り組みを支援

CRA (サイバーレジリエンス法) は、製造業者に対し、セキュア・バイ・デザインの原則を適用し、製品ライフサイクル全体を通じて脆弱性を主体的に管理することを義務付けています。Axivion の静的解析チェッカーは、CWE (Common Weakness Enumeration) および CERT セキュアコーディング規格に対応しており、バッファオーバーフローやレースコンディション (競合) などのセキュリティ関連の弱点を開発初期段階で特定することを支援します。

また、Axivion のアーキテクチャ検証機能は、コードベースの構造的な健全性と保守性の長期的な維持をサポートします。これは、「製品のサポート期間全体を通じてセキュリティを維持する」という CRA の要件に対応するものです。これらの機能により、Axivion は CRA が求めるセキュアなソフトウェア開発プロセスを実現するための有効なツールとして位置づけられます。

# 主要機能

## アーキテクチャ検証

- Integrated modeler
- Interfaces to UML tools
- AUTOSAR XML import (ARXML)
- Freedom from Interference
- Architecture reconstruction
- Architecture views for safety and security
- Architecture Export
- Architecture as Code

## コーディングガイドライン

- NVIDIA CUDA C++ ガイドライン
- MISRA C/C++
- AUTOSAR C++14
- SEI CERT® C/C++
- C Secure Coding
- CWE
- Qt-specific rule sets
- Best practice (e.g. CQM)
- Customized checks

## メトリクス監視

- HIS
- OO Design metrics
- Complexity metrics
- Best practice and customized checks

## デッドコード分析

- Reachability analysis
- Analysis of libraries

## クローンの検知と管理

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

## 循環依存検知

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

## 不具合検知

### メモリとポインター

- NULL参照
- ローカル変数のエスケープアドレス
- new/mallocからのメモリリーク(delete/freeなし)
- リソースの割り当て/解放の不一致
- 同時に読み書きに使用されるリソース
- リソースに対する禁止された操作
- 二重解放/解放後の使用
- 関連しないポインターの比較/減算
- 配列の範囲外アクセス
- バッファオーバーフロー
- 副作用
- 汚染解析
- レースコンディション解析

### 例外

- スタック戻し時の例外
- 例外規定の違反
- キャッチされなかった例外
- 到達しないキャッチブロック

### 数的エラー

- ゼロ除算
- 算数演算におけるオーバーフロー
- 列挙型変数への不正な値の代入

### 論理エラーとカスタマイズされたルール

- 禁止された引数の値
- 未初期化変数
- 未使用の定義
- 定数条件



#### Note

Axivionツールは主に実行時エラーに焦点を当てていますが、APIの使用方法や構文エラーなど、その他の問題も網羅しています。

詳しくは、お気軽にお問い合わせください。

# 基本仕様

Axivion 7.12 の機能の一部を抜粋しています。

機能詳細についてはお問い合わせください。

## 解析対象言語とサポートコンパイラ

言語	C, C++, CUDA C++, C# <sup>1)</sup> , RUST <sup>1)</sup>
コンパイラ	Blackfin, Clang, Codevision, CodeWarrior®, Cosmic, Green Hills Software®, GNU, IAR™, Keil™, Microchip®, Microsoft®, NVIDIA nvcc, Renesas, Tasking, TI, Windriver, Others

## プラットフォームOS

Host OS	Windows® 10/11 or Windows® Server® 2016/2019/2022/2025 in 64bit x86_64 GNU/Linux® (minimum requirement is glibc2.28 or later), Linux ARM64 macOS® (minimum requirement is macOS 14 in 64 bit), macOS® ARM64
---------	---

## プラグイン

IDE	Qt Creator, CLion, Eclipse™, Eclipse-based (e.g. e <sup>2</sup> studio, 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

## 対応バージョン管理システム

バージョン管理システム	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®
-------------	--

## 対応 UML® ツール

UML® ツール	IBM Rational Rhapsody, Sparx Enterprise Architect (via XML or .qea-files), PlantUML
----------	---

## 対応コーディングガイドライン

コーディングガイドライン	NVIDIA CUDA C++ Guidelines, MISRA C:2004/2012/2019/2023/2025, MISRA C++:2008/2023, AUTOSAR C++14, CERT C Rules, CERT C Recommendations, CERT C++ Rules, CWE, ISO / IEC TS 17961, Qt Framework, Customized checks and other
品質ガイドライン <sup>2)</sup>	ISO 26262 up to ASIL-D IEC 61508 up to SIL-4 IEC 62304 up to Class-C EN 50128, EN 50628

## その他

対応ブラウザ	Microsoft® Edge, Mozilla Firefox®, Google Chrome™
要件	Java® Runtime (17,21,25)
アドオン	C/C++/CUDA C++向けAxivionツール認定キット(静的コード解析およびアーキテクチャ検証)

<sup>1)</sup>Limited support

<sup>2)</sup>Certified by SGS-TÜV Saar

技術データは予告なしに変更されることがあります。無断複製・転載を禁じます。すべての会社名および/または製品名は、各市場および/または各国における各メーカーの商標および/または登録商標です。弊社は常に最新のデータ状況をパートナーにお届けするよう努めています。製品リリース時期と本ドキュメントの公開時期の間に、仕様変更される可能性があります。



お問い合わせ

Qt Group 日本オフィス

〒100-0005 東京都千代田区丸の内3-3-1 新東京ビル4F

Tel: 03-6264-4500 Mail: [japan@qt.io](mailto:japan@qt.io)

Qt Group (Nasdaq Helsinki QTCOM) はグローバルなソフトウェア企業です。産業界のリーダーと 150 万人を超える世界中の開発者が信頼を置き、ユーザーに愛されるアプリケーションやスマートデバイスを作成しています。UI デザインやソフトウェア開発から品質管理と導入まで、製品開発サイクル全体を通してお客様の生産性向上を支援します。Qt Group のお客様は 70 以上の業界で 180 か国以上に広がっています。Qt Group の従業員数は世界で約 1,100 名、2025 年の売り上げは 2 億 1,630 万ユーロでした。