

NVIDIAのCUDAを活用する  
安全クリティカルなシステム向け

# GPU アクセラレーションか、 それとも制御不能な コンプライアンスリスクか？

機能安全におけるリーダーへの警鐘

## このガイドの対象

### 瀬戸際に立つエンジニアリングリーダーへ

あなたは重要なGPUアクセラレートシステムを担っています。チームは病気を診断するAI、車を走らせるAI、産業機器を制御するAIを作っています。賭かっているのはお金ではなく人命です。

現在の進め方は持続不可能です。最高のエンジニアが時間のほぼ半分を機能安全システムの手作業検証に費やしています。コンプライアンスがボトルネックになり、GPUのバグ1つで単なるリコール以上の事態を引き起こしかねません。

このガイドは、「速く出荷するには存亡リスクを受け入れるしかない」という考えを拒み、開発チームを「何とかやり過ごす」状態から解放したいリーダーのためのものです。

### このガイドがロードマップになるのは次のような場合です：

ツールがCUDAを理解できずGPUコードを検証できない

手作業検証が加速効果を食いつぶし、GPU投資が報われていない

システム障害の責任を負い、「全力を尽くした」は法的に通用しない

革新的なプロダクトを作るために入社した人材が、スプレッドシートを維持するだけの日々に失望して離職している

### このガイドはすべての人に向けたものではありません

#### 対象外：

手作業検証で満足している組織

責任は自分たちに降りかからないと考えるリーダー

運を頼みに成功を賭ける企業

#### 対象：

安全クリティカルシステムのバグ1つが重大な結果をもたらすと理解しているCTO・エンジニアリングVP

評判と法的リスクがシステム安全に依存することを理解している技術リーダー

責任と予防コストの計算を済ませているエンジニアリング幹部

「安全クリティカルなコードを全て検証できなかった」では済まないと知る意思決定者

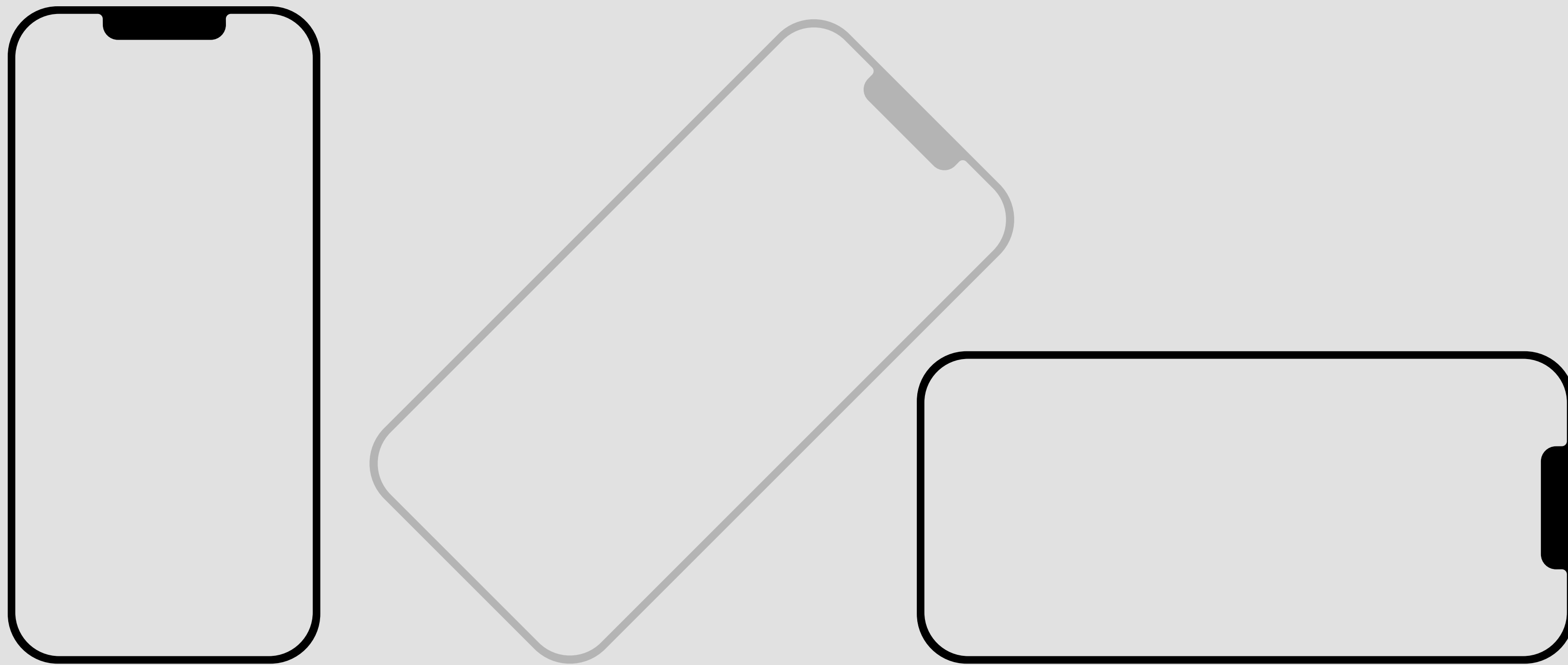
### なぜ今なのか

裁判所は数億ユーロ/ドル規模の制裁金を科し、ソフトウェアエラーで企業は停止しています。これは警告ではなく、今まさに積み上げられている前例です。

このガイドは、明日の安全でセキュアなソフトウェアを作るためのコンプライアンス基盤の構築方法を示します。



モバイル端末で読んでいますか？  
**横向きにすると読みやすくなります。**



目次

1. 安全クリティカルなGPU投資に潜む不都合な真実	05	6. 理論と現実が交わる場所	24
2. もはやスケールしない卓越性	08	7. 機能安全と CUDA : 不可欠な要件	27
ソフトウェアエロージョンは逆向きの複利	11		
3. 誰も語らない二層の問題	12	まとめ	30
4. 構造的コントロールレイヤーの構築	17	すでに市場は二分されている	31
変革の 3 フェーズ	19	選択の瞬間	32
		チャンスの窓は長く開かない	33
5. CFOを納得させるROI	20	GPU 検証基盤を構築する準備はできていますか	34
隠れたコストと将来像	22		

# 1

## 安全クリティカルな GPU 投資に潜む不都合な真実

# 安全クリティカルな GPU 投資に潜む不都合な真実

命を救う医療診断AI、事故を防ぐ自動運転、ヒトと協働する産業ロボットなど、あなたの組織は、人命に関わるプロダクトを作っています。

それらは機能安全要件を持つ安全クリティカルシステムで、生死を左右します。もはや「単なるソフトウェア」ではなく、責任そのものです。

優れた機能安全製品には、より大きな責任とリスクが伴います。

「優れた機能安全製品には、  
より大きな責任とリスクが伴う」

投資は巨額でした。GPU、ライセンス、トレーニング、スペシャリスト。現代のソフトウェア開発はGPUアクセラレーションに将来を賭けています。その決断は、最高水準の安全を満たしつつ産業を変革するAI性能を約束していました。

しかし、予算にサインした瞬間からリスクも伴っていました。

安全クリティカルコードの約 40% が GPU 上で動作し、一般的な検証ツールには見えません。「見ない」のではなく「見えない」のです。静的解析はCUDAカーネルを暗号文のように扱います。本来は並列実行パターンやメモリ階層、無数のスレッドが形作る振る舞いを見るべきところを、単なるテキストとしてしか見ていません。

規制の厳しい業界では、すべての CUDA カーネルを検証し、ドキュメント化し、安全要件にトレースしなければなりません。AIを搭載する自動車はISO 26262 ASIL Dに準拠する必要があります。医療システムはIEC 62304、産業オートメーションはIEC 61508が求められます。それでも多くの組織は手作業で、受け身で行い、自動化すべき作業に最高のエンジニアを消耗させています。

技術的な課題に見えても、影響はチームや業務、市場投入までの時間に及びます。

避けられない事故が起き、GPUバグが原因となれば、法廷で最初に問われるのは「安全確保のため利用可能な手段をすべて使ったか」。ISO 26262 は最善手段の利用を義務付けています。NVIDIAのCUDA Safety-Critical Programming ガイドライン (\*1) 以前はグレーでしたが、今は違います。

これは現代のエンジニアリング組織を締め付けるパラドックスです。開発を加速するはずの技術が、最大のボトルネックであり最大の責任源になっています。

しかも日々、複利のように悪化します。

<sup>1</sup> CUDA Safety-Critical Programming ガイドライン ➤

# 2

もはやスケールしない  
卓越性



# もはやスケールしない 卓越性

ここまで来たのは偶然ではありません。数十年にわたり世界水準の製品を築き、プロセスで障害を防ぎ、精緻に実行してきました。なのに、なぜGPUアクセラレート版のリリースごとに「バグ1つで破滅」の感覚があるのでしょうか。



理由は「これまでの作り方」と「GPUアクセラレートシステムの実際の動き」のミスマッチ にあります。従来の検証は逐次実行を前提に、関数を1つずつレビューし経路を追う前提でした。しかしGPUカーネル1つで数千のスレッドが立ち上がり、それぞれが複数のメモリ階層に触れます。レースコンディションは実行のたびに変わるタイミングパターンに潜みます。

日々、最高のエンジニアが直面するのは、製品を差別化するAI機能の開発ではなく、人間デバッガーとして自動で検証すべき経路を手作業で追うことです。スプリント前に陳腐化するカーネル依存のスプレッドシートを更新し、今月3回目のレビュー会で特定のメモリアクセスが安全な理由を説明しています。

彼らは複雑さに苦しむジュニアではありません。貴重なCUDAエキスパートであり、本来自動検証すべき作業に、作業時間全体の30〜40%を費やしています。

経済性だけでも脅威ですが、戦略的な影響はもっと深刻です。

# ソフトウェアエロージョンは逆向きの複利

チームが浪費している時間は、GPUシステムで「逆複利」のように働くソフトウェアエロージョンを防ぐ ことに消えています。

## 技術的負債の増幅

未検証のカーネルはリスクを「足す」のではなく「増幅」します。CUDAでの安易な近道やパッチは並列性ゆえに複利的に積み上がります。

## 依存の転移

GPUカーネルは安全境界をまたぐ隠れた依存を抱えます。「AIモジュールだけ」のつもりがシステム全体に広がります。

## 知識のボトルネック

GPUコードとアーキテクチャを理解するのはごく少数。彼らが去れば埋め難い知見も去ります。アーキテクチャは頭の中にあり、かつて検証された構造にはありません。

## ドキュメントの劣化

カーネル挙動の手書きドキュメントは完成前に陳腐化します。アーキテクチャの真実は属人的な暗黙知になっていきます。

## 責任のエスカレーション

未検証コンポーネントは露出を増やします。例: 安全クリティカル部品で責任を負う Tier-1 サプライヤーは、数千万ドル規模のリコールリスクを抱えます。この責任リスクを保険で賄うのに毎年何百万ドルもかかります。事故が起これば保険はブランドを回復してくれません。

結果として、競合が四半期ごとに出荷する間、あなたは、昨年版の検証と火消しに追われ続けます。

# 3

## 誰も語らない 二層の問題



# 誰も語らない二層の問題

多くの組織が誤るのは、GPU検証をツールの問題として扱うことです。より良い静的解析を買う、レビューを増やす、プロセスを追加するーこれらは病気を放置したまま症状だけを扱っています。

真の問題はほとんど哲学的です。昨日のマインドセットで明日の技術に臨んでいる。既知のプロセスで制御できるとみなす考え方です。コードがよりシンプルで逐次実行し、熟練エンジニアがモジュール全体を頭に入れられた時代には機能しました。

GPUアクセラレートシステムを展開する組織は、  
**2つの層**で重大な検証ギャップに直面しています：

第一層 AI モデル

訓練し、検証し、テストできる。  
何らかのプロセスがあり、  
今のところ何とか回している。

第二層 GPU インフラ

AIを支えるCUDAコードは  
検証の空白地帯にある。  
従来ツールは解析できず、手作業レビューは  
数千の並列スレッドにスケールしない。

このギャップは組織の断絶を生みます:

リーダー陣は正当な恐れから動く

すべてを検証しきれないAIシステム、  
未検出バグによる責任、  
組織の非効率、より速く出荷する競合、  
巨額のGPU投資のROI証明。

エンジニアは日々の苛立ちから動く

自動化されるべき手作業検証、  
到着時点で陳腐化したドキュメント、  
適切なツールなしで安全性を示さねばならない現実、  
時間的プレッシャー。

皮肉な現実とは？

AIモデルが完璧でもCUDAインフラが壊れればシステム全体が壊れます。カーネルのレース  
コンディション1つ、特定タイミングで破綻するメモリアクセスパターン1つで、数か月のやり直  
しになります。

これはレアケースではなく、GPU環境に内在する系統的风险であり時限爆弾です。経営の  
恐れにプロセスを、エンジニアの苛立ちにツールを、という従来の処方箋は病気を長引かせ  
るだけです。壊れた基盤の「対処」に人を縛りつけ、直すことを妨げます。



# 4

## 構造的コントロールレイヤー の構築

# 構造的コントロールレイヤー の構築

持続不能な混乱から統制へ変わるには、根本的な事実を受け入れる必要があります。GPUアクセラレートシステムのために設計された検証基盤が要るということです。

これはAIの野心とGPUハードの間に欠けているレイヤーを作ることだと考えてください。全てのCUDAカーネルを検証し、安全境界を強制し、コンプライアンス要件を自動で満たす構造的コントロールレイヤーです。

官僚的な重荷を増やすものではありません。エンジニアを押し潰し、ビジネス成長を妨げる手作業の負担を取り除きます。



# 5

## CFOを納得させるROI



# CFOを納得させるROI

コストについて率直になりましょう。どの組織も既にGPUガバナンスに支払っていますが、最も高価な通貨——時間、人材、市場ポジション——で払っています。

今の現実を計算してみましょう(次ページの図解を参照)。

3人のシニアエンジニアが、業務時間の3分の1を手動検証に費やしている——これは実質的に、1人分の人員がドキュメント作成だけに従事しているのと同じです。その総コストを、GPUコードを扱うチームの数だけ掛け算してください。さらに、実装されなかった機能、進められなかったイノベーション、参入できなかった市場という機会損失も加えてみてください。

さらに監査準備を考えてみてください。四半期ごとに3〜4週間、チーム全員が生産的な業務を止め、本来自動で検証されるべきコードを手作業で追います。この遅延の連鎖、守れないコミットメント、失われる機会は、四半期ごとに積み重なっていきます。

最大のコストは人材かもしれません。優秀なエンジニアは、コンプライアンス担当になるために入社したわけではありません。未知の領域を切り拓くためです。火消しに費やす日々が続けば辞める日が近づき、代えのきかない知見も失われます。

本当のリターンは、新しく開けるチャンスや、できることの広がりにあります。



隠れたコストと将来像

現状（いま払っているもの）

シニアエンジニア3人 × 30%の手作業検証 = 1名分のヘッドカウントがドキュメント要員に

監査準備: 四半期ごとにチーム全員で4週間 = 50万ドル超の生産性が凍結

不満で離職したエンジニアの補充 = 6〜12か月+失われる知見

本番に届くGPUバグ = 数百万ドルのリコールリスク

責任リスクの保険 = 毎年数百万ドル

安全事故によるブランド毀損 = 計り知れない

将来像（適切な基盤/構造的コントロールを持った場合）

手作業検証が50%減 : 数か月が数日に

監査準備は70〜80%短縮 : 3〜4週間が自動レポートで数時間に

開發生産性は15%向上 : 開発者1人あたり年2〜3か月を取り戻す

CPUとGPUで100%の検証カバレッジを持つことで コンプライアンスが競争優位に

デバッグは30〜40%高速化 : 本番ではなくコミット時に問題を捕捉

責任防止 : 安全保証に「可能な限り最善の手段」を示せる

真の価値はコスト削減ではありません。「我々のAIシステムは速く、安全で、設計上証明可能です」と顧客、規制当局、投資家に言えることです。それはコンプライアンスやおまけではなく**競争優位**です。

事故やエラーが起きたとき、利用可能な手段をすべて用いたと証明できます。それが、対処可能なインシデントと**会社を揺るがす責任**の差になります。

# 6

## 理論と現実が 交わる場所



# 理論と現実が交わる場所

これは仮定ではありません。  
業界をまたいで繰り返されるパターンです。

医用画像と手術ロボット

あなたのCT/MRI再構成はCUDAで動き、手術ロボットはGPUでリアルタイムの画像認識と制御を行います。モデルは完全に検証済みでも、基盤のCUDAカーネルにあるレースコンディションで断続的な不具合が起こり、診断ミスや手術動作の遅延で初めて発覚します。

CUDAプラットフォームで開発する医療ソフトウェアチームは、IEC 62304に沿った安全なコーディングを守る必要があります。適切な検証なしでは、診断AIを支えるGPUコードを検証したのかと問われる医療過誤訴訟がバグ1つで発生します。

自動運転車とADAS

SDVの安全クリティカルコードの最大40%がCUDAベースです(センサーフュージョン、認識スタック、意思決定アルゴリズムなど)。GPUで加速したセンサーフュージョンはテストでは完璧でも、特定タイミングでメモリ破壊が起き、AIが障害物を見逃すケースを手作業テストは捕捉できませんでした。

CUDAコードを未確認のままにはできません。ISO 26262 は現実的な強制力を持ち、「GPUコードを検証するツールがなかった」はもはや言い訳になりません。

産業・ヒューマノイドロボット

協働ロボットはGPUで認識・経路計画・制御を行います。人検知のコンピュータビジョンは全テストを通過しても、未検証のカーネル相互作用が安全応答時間を乱し、人が作業領域に入ったときに間に合わないことがあります。

タイミング違反1回、人検知の見逃し1回で、リコールだけでなく責任追及に直面します。規制当局からの質問はまたシンプルです。

パターンは一貫しています。AIモデルが完璧でも、基盤のGPUインフラが正しく検証されていなければ壊滅的に失敗し、**その責任はあなたに降りかかります。**

# 7

## 機能安全とCUDA 不可欠な要件



# 機能安全とCUDA 不可欠な要件

NVIDIAが 2025年10月 にCUDAコーディングガイドラインを公開したとき、規制産業が痛みをもって学んだことを認めました。「今日、GPUはもはやグラフィックスやHPCにとどまらず、安全クリティカルシステムを可能にするアクセラレータになっている」。





# まとめ

# すでに市場は二分されている

2つの陣営が生まれ、ギャップは四半期ごとに拡大しています。

違いはリソースや人材ではありません。昨日のプロセスが明日の技術を扱えないと認めたかどうかです。

規制がGPU検証を求めるとき、陣営2はすでに準拠しています。  
法的に安全性の証拠を求められるとき、陣営2はエビデンスを持っています。  
監査が必要なとき、陣営2は慌てません。

陣営1は右往左往し、陣営2は備えています。ギャップは四半期ごとに広がります。  
どちらの陣営に入りますか？

手作業レビューを増やす

検証担当を増やす

手作業レビューを増やす

陣営 1

人材が流出する

競合が出荷する間、遅延を説明する

事故が起きたとき、GPUコードを検証できなかった理由を説明する

GPUガバナンスを体系化する

他社がデバッグする間に四半期ごとに出荷する

ドキュメントではなく革新の場を提供し、優秀な人材を惹きつける

陣営 2

事故が起きたとき、安全確保に全手段を尽くしたと証明する

# 選択の瞬間

選ぶのはツールやプロセスではなく、**2つの未来**のどちらかです。

複利効果が決めるのは市場ポジションだけでなく生存です。他社がAI機能の遅れを弁明している間に、あなたは出荷します。他社が人材を失う間に、あなたは人材を惹きつけます。他社が監査や訴訟を恐れる間に、あなたは自らの手法を証明の機会として歓迎します。

## 耐えるだけ

重大な問題を見逃す手作業レビューの増加

取締役会で繰り返される遅延説明

## 未来 1

離職する人材の増加

避けられないGPU関連事故が起きたとき、なぜ検証できなかったかを説明する

裁判所も顧客も同情しない

## 基盤を築く

GPUアクセラレーションを戦略的優位に変える

継続的かつ自動の検証能力を持つ

## 未来 2

エンジニアを解放し、本質的な製品開発に集中させる

事故が起きたとき、合理的なあらゆる手段を講じたと証明する

品質を「コスト」から「鎧」へ変える

# チャンスの窓は 長く開かない

GPU革命は約束から必須条件へと移りました。あらゆる安全クリティカル産業が、市場をリードするAI機能のためにGPUアクセラレーションに依存しています。

勝者と敗者を分けるのは、AIを動かすGPUコードが安全・準拠・法的に防御可能であることを証明できるかどうかです。

エンジニアは自分たちが作るAIシステムを誇りに思っています。人命を救い、災害を防ぐ製品を作っています。しかし適切な安全クリティカル検証基盤がなければ、手作業に知見を浪費し、イノベーションが止まり、組織は存亡リスクにさらされます。

標準は整備されつつあり、勝者は生まれつつあり、責任の前例は積み上がっています。

適切なGPUガバナンスがなければ、日々リスクは複利で膨らみます。

→ 手作業検証に浪費される生産性

→ 検証ボトルネックによるイノベーションの遅延

→ 課題を解決した競合に譲る市場

→ 未検証カーネルが増えるほど拡大する責任リスク

GPUアクセラレートシステムのために構造的コントロールレイヤーを築く組織は、標準に「従う」だけでなく、それを「定義」します。AIを「出荷」するだけでなく、自信と法的防御力を備えて出荷します。

GPU 投資は本来、画期的なAI機能を支えるためのものでした。適切な検証基盤なしでは、リスクと責任しか生みません。正しいコントロールレイヤーを持てば、本来の目的——信頼でき、安全で、将来に備わった製品の土台——になります。

GPUガバナンスが必要かどうかではありません。競合より先に構築するかどうかです。

# GPU検証基盤を構築する 準備はできていますか？

課題と、事の重大性は理解できました。次は行動です。

Axivion for CUDAは、NVIDIAのCUDAコーディングガイドラインと主要な安全標準をサポートする、GPUコードとアーキテクチャ検証における業界初かつ唯一のソリューションです。

私たちは、コンプライアンスをボトルネックから競争優位へ変えるリーディング組織を支援します。

## Axivion for CUDA

[qt.io/quality-assurance/axivion-for-cuda](https://qt.io/quality-assurance/axivion-for-cuda)





# Qt Groupについて

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900名、2024年の売り上げは2億910万ユーロでした。

詳細は [www.qt.io/ja-jp](https://www.qt.io/ja-jp) をご覧ください。