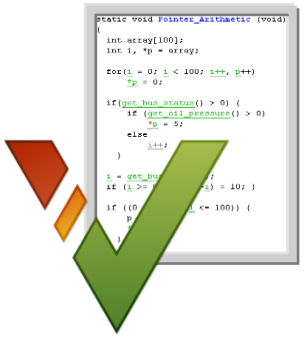
JaSST'18 Tokyo

2018/03/07（水）



静的解析とチーム間レビュープロセス

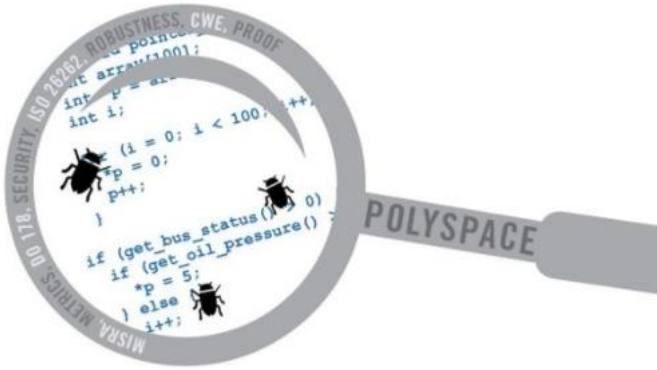
～Polyspace®アドバンスト静的解析の有効活用～

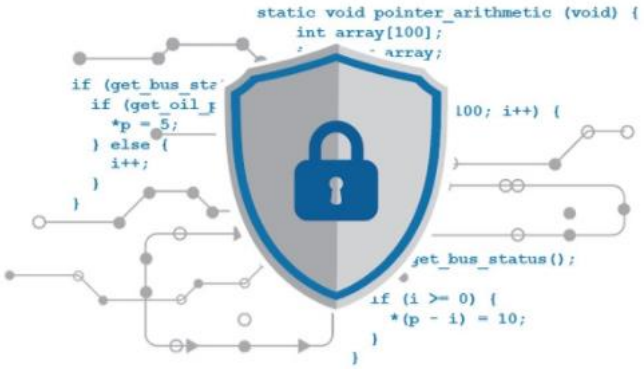
MathWorks Japan

Application Engineering

アジェンダ

- イントロダクション
- Polyspace静的解析の特徴
- 静的解析に関する課題と解決策
- ユーザ事例 & 最後に. . .

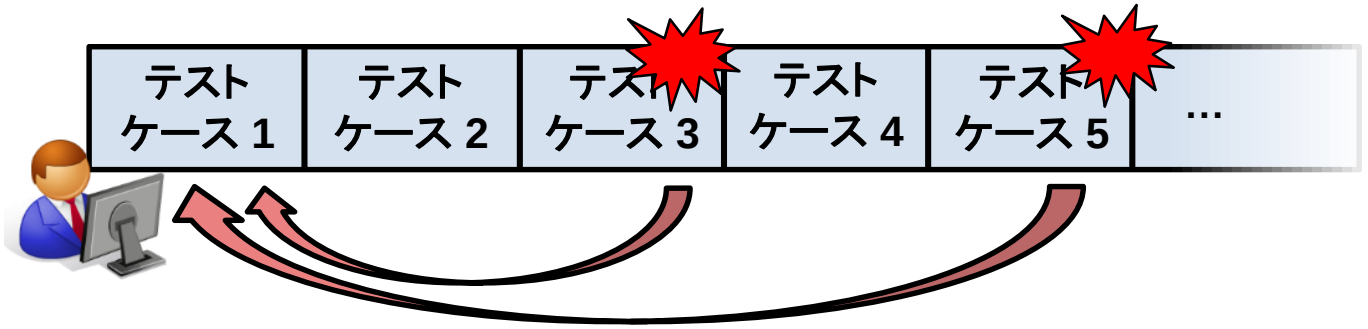




動的テスト実行に関する課題

動的テストは動作確認で必要だが、
エラー検証には非効率

従来のソフト検証手法：動的テスト



ユニットテスト
&
統合テスト

問題点：

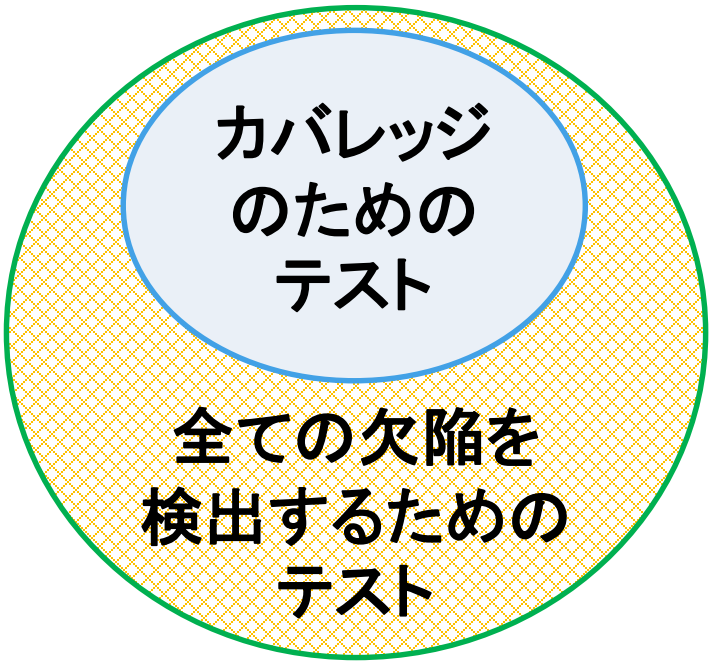
エラー検出の遅れ　　納期の延期　　リワーク

多数の繰り返し作業　　納品物でエラーが発生

ソフトウェア品質確保の効率化に向けて

提案：静的解析によるソフトウェア品質の確保

- 従来的の動的テスト手法
  → テストケースの作成、実行、確認が必要




```
x / (x - y)
```

- 未初期化変数
- オーバーフロー
- ゼロ除算

`int32` の場合、4.61×10^{18} のテストケースが可能

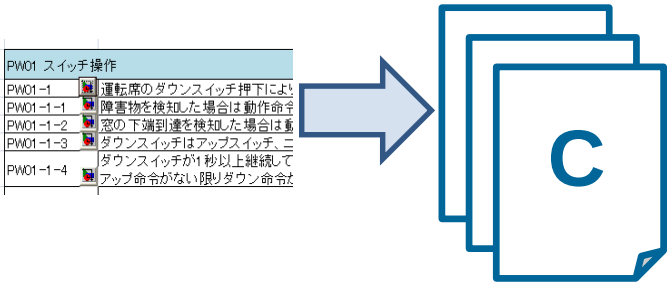
全て可能なテストケースの作成・実行・レビューは不可能！

ソフトウェア検証のストリームライン

検証目的により適切な検証手法を使用して
全体プロセスを効率化

動的テスト

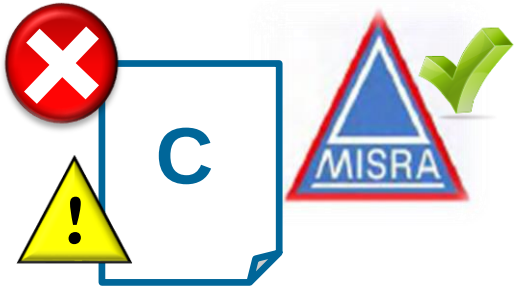
- 要求仕様通りの設計かを確認
- 要求仕様に問題がないことを確認



機能チェック

静的解析

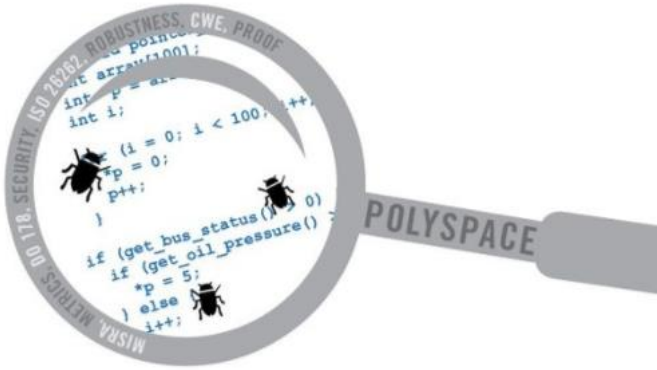
- エラー・欠陥の存在を確認
- スタンダード準拠の確認

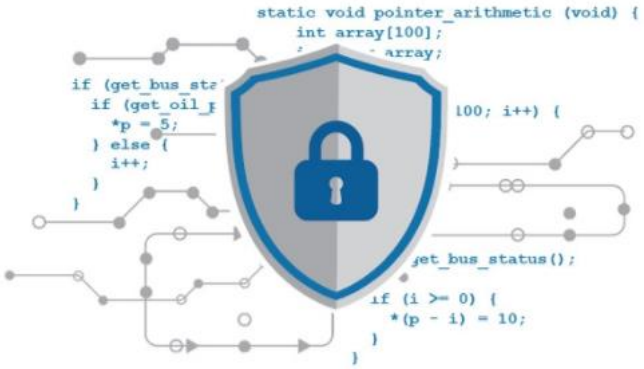


ソフト品質

アジェンダ

- イントロダクション
- Polyspace静的解析の特徴
- 静的解析に関する課題と解決策
- ユーザ事例 & 最後に. . .





Polyspace：ソフトウェア静的解析ソリューション

コードの安全性とセキュリティの確保を支援

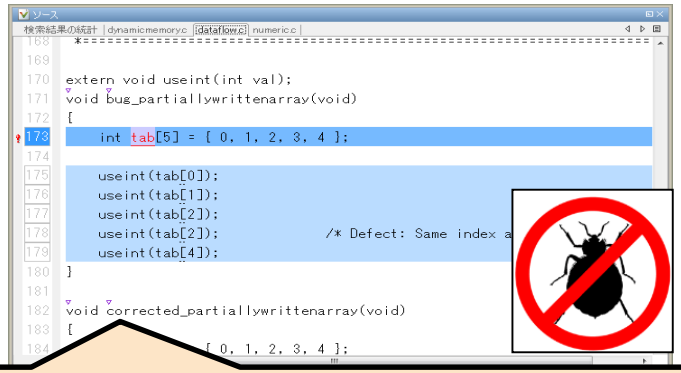
Polyspace Bug Finder™
バグ・セキュリティ脆弱性の検出

Polyspace Code Prover™
安全性の証明

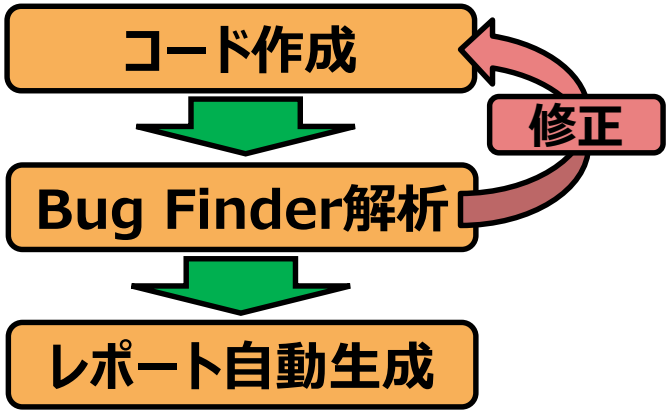
Polyspace Bug Finder のスピーディー解析

素早い欠陥検出・修正を可能とする！

- **セキュリティ脆弱性・バグ検出**
  - コード作成後、すぐに欠陥を検出・修正
- **コーディングルールチェック**
  - エラー予防と再利用性向上
  - MISRA準拠を確認
- **コードメトリックス解析**
  - コードの複雑度を測定



時間を掛けずに大部分のバグを識別



開発プロセスの上流で不具合を発見！
コードを統合前に多くの欠陥を修正！
コード開発の効率に繋がる！

セキュリティ脆弱性・バグ検出項目

数値
- ゼロ割、オーバーフロー
- 標準ライブラリ数学ルーチン
  の無効な使用
- …

静的メモリ
- 配列の範囲外アクセス
- NULLポインター
- …

プログラミング
- 等号演算子による浮動小数点の比較
- 宣言の不一致
- …

動的メモリ
- メモリリーク
- 前に解放したポインターの使用
- …

セキュリティ
- パス操作が脆弱
- 疑似乱数発生器が脆弱
- …

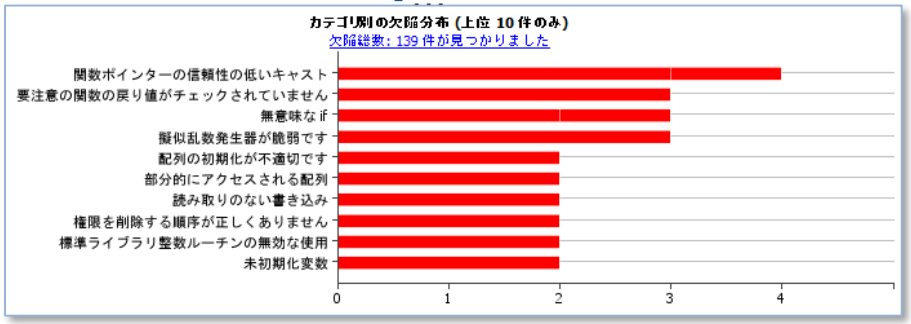
汚染されたデータ
- 汚染された文字列形式
- 汚染されたサイズでのメモリの割り当て
- …

データフロー
- 読取りのない書込み
- 未初期化変数
- …

同時実行
- データレース
- デッドロック
- …

リソース管理
- 読取り専用リソースに書込み
- 以前に閉じられたリソースを使用
- …

適切な手法
- 未使用のパラメータ
- 値渡しの大きな引数
- …



https://jp.mathworks.com/help/bugfinder/defect-reference.html

サイバーセキュリティ： 業界活動と標準

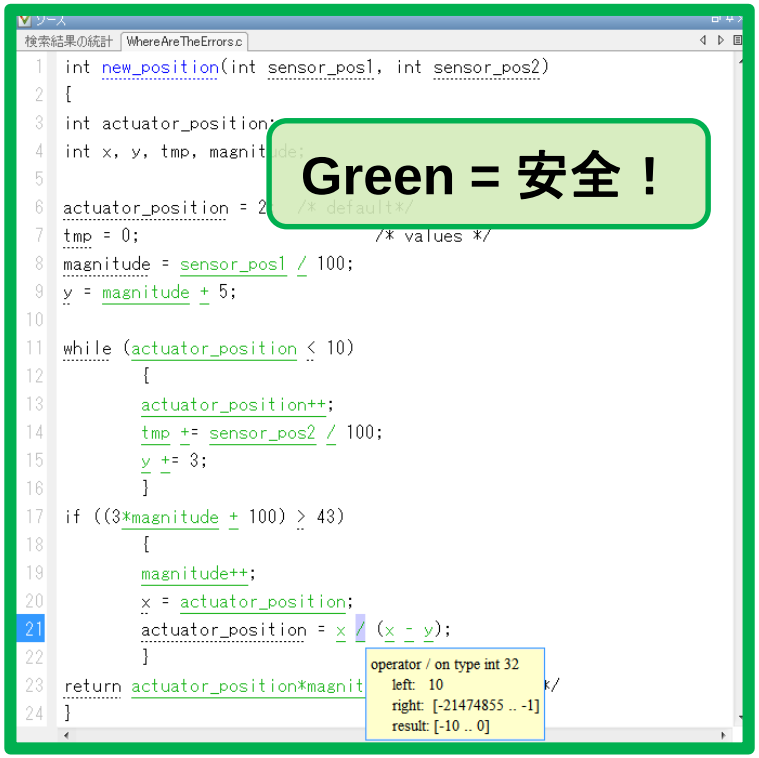
最近注目のコーディング標準＆実践

- **CERT C**：セキュアコーディングスタンダード
- **ISO/IEC TS 17961**：Cセキュアコーディングルール
- **CWE**：共通脆弱性タイプ
- **MISRA-C:2012** Amendment 1：改訂1

ソフト安全性：Polyspace Code Proverのコード証明

Polyspace：品質確保の上、検証工数の削減に貢献

- Polyspaceはコードの安全性を確保するための静的解析を提供
- コードの品質を証明することにより、検証プロセスの効率化に繋がります

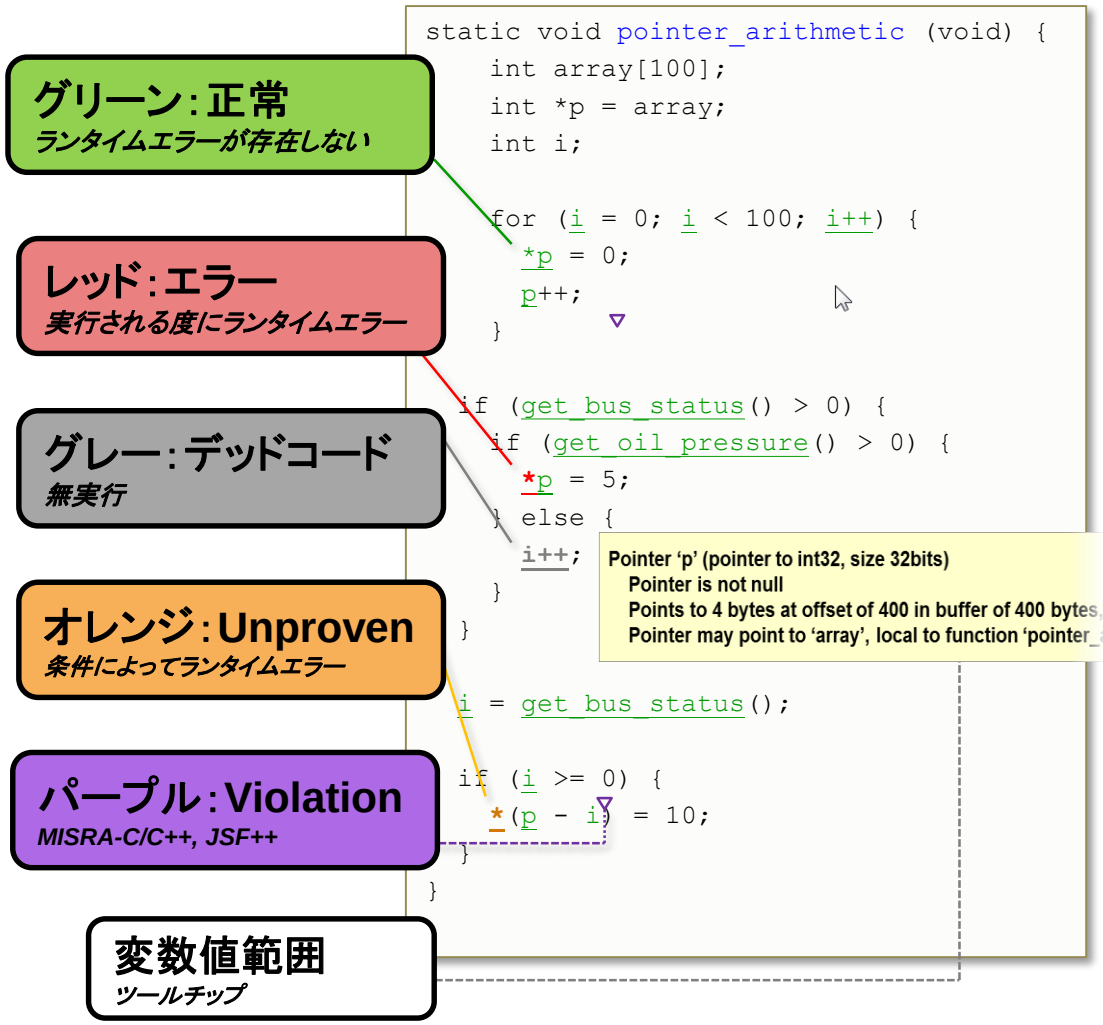


Polyspace Code Prover でコードの正しさを証明

全ての実行パス・変数レンジの結果を考慮する！

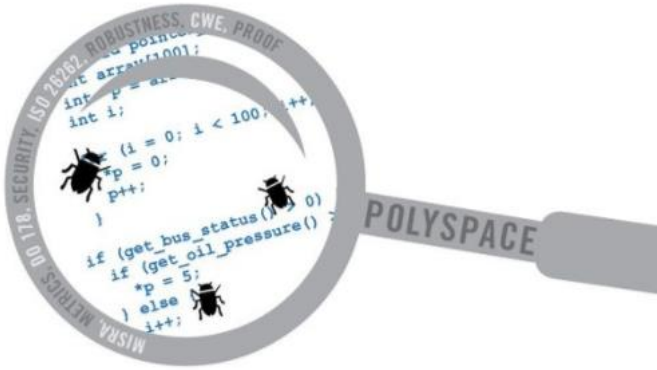
- **Quality（品質）**
  - ランタイムエラーの証明
  - 測定、向上、管理
- **Usage（使用方法）**
  - コンパイル、プログラム実行、テストケースは不要
  - 対応言語：C/C++/Ada
- **Process（プロセス）**
  - ランタイムエラーの早期検出
  - 自動生成コード、ハンドコードの解析可能
  - コードの信頼性を測定

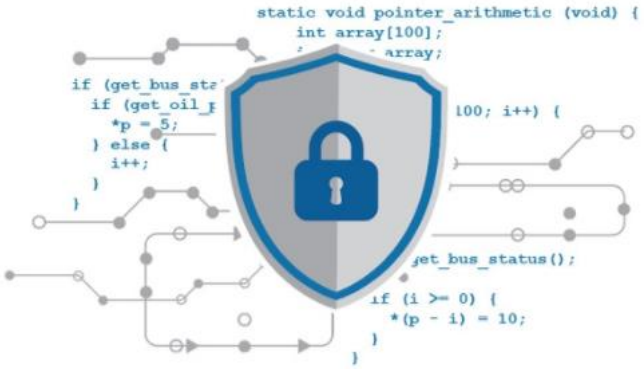
実機実験前にコード信頼性を確保して手戻りを削減



アジェンダ

- イントロダクション
- Polyspace静的解析の特徴
- 静的解析に関する課題と解決策
- ユーザ事例 & 最後に. . .





静的解析をご活用する際のチャレンジ

ツール担当者側

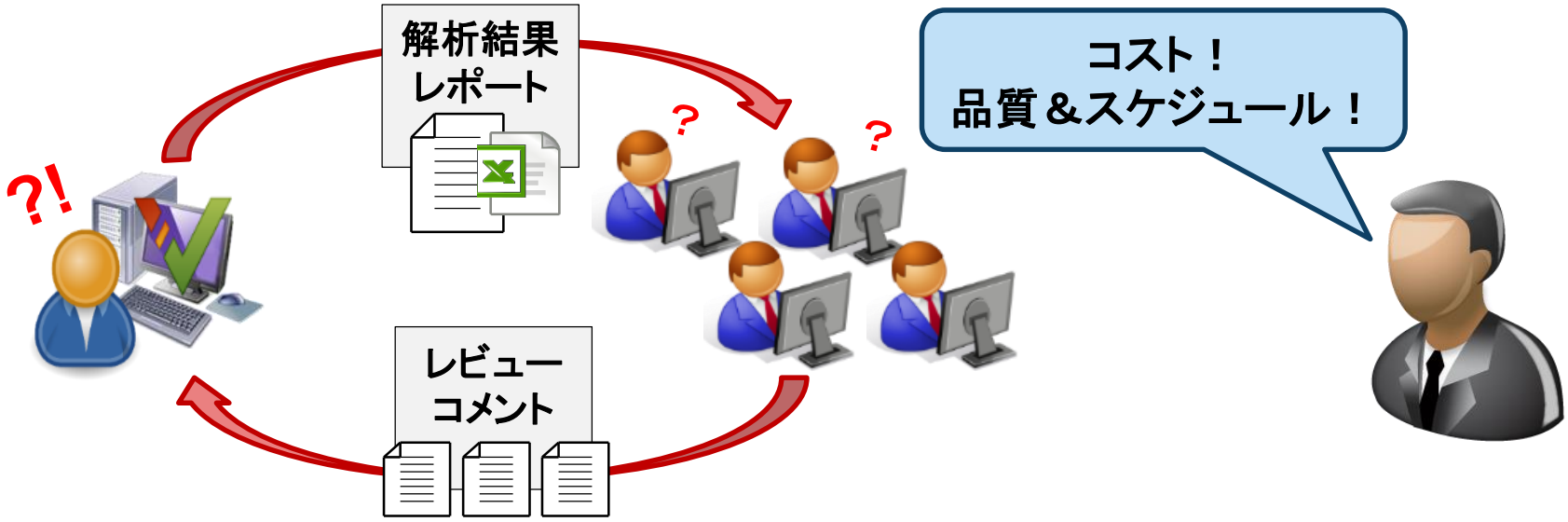
- 毎回解析の手動実行が手間
- 解析結果を確認する方が少ない
- レビューコメントの統合が困難

デベロッパー側

- 静的解析ハードルにより活用しない
- 解析結果の取得方法・タイミングが困難

マネージメント側

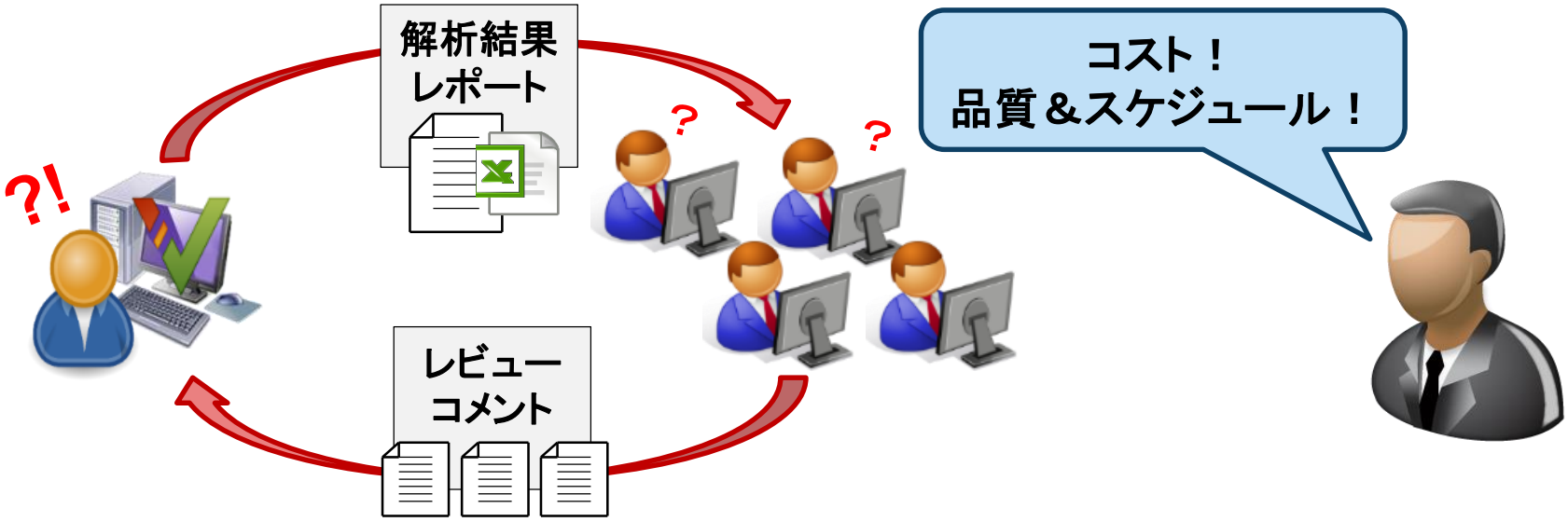
- ライセンス利用の効率化が必要
- ツール利用者が少ないためバグが発生



静的解析をご活用する際のチャレンジ

非効率なマニュアル作業

ツール実行　　結果共有　　レビュー管理　　情報シェア　　ツール管理

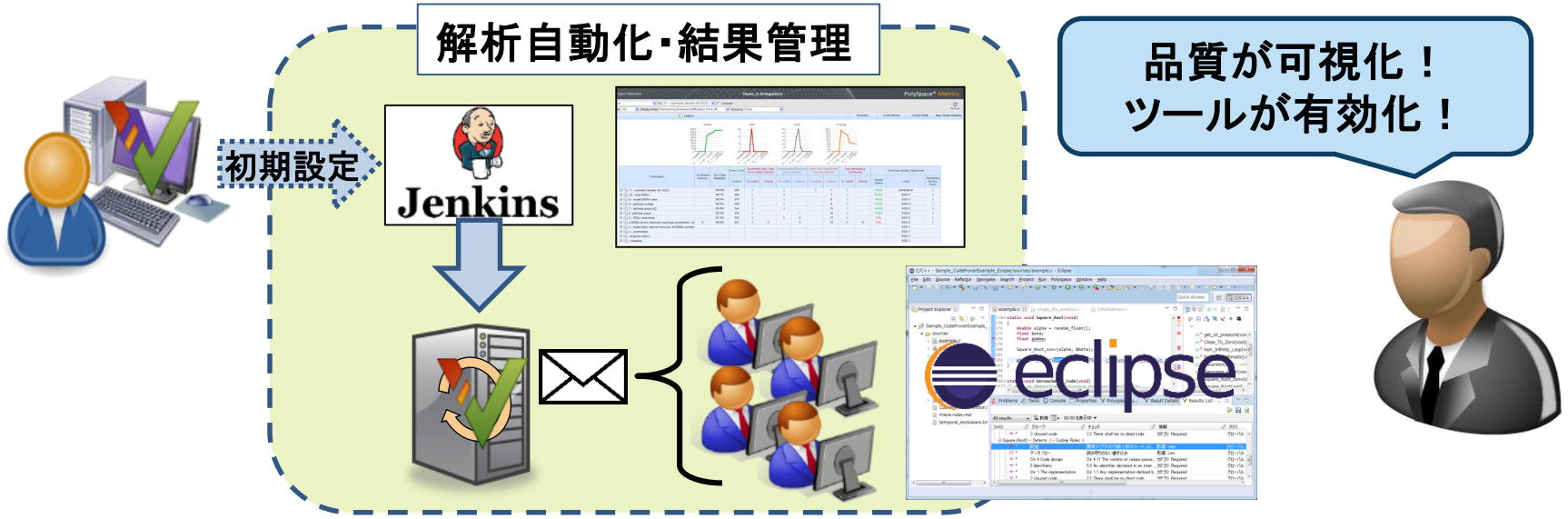


静的解析プロセスの効率化

プロセスを可能な限り自動化
有効ツール機能を利用した効率化

CIツール連携　解析の自動実行　情報の自動発信

結果の一括管理　無駄ないライセンス利用



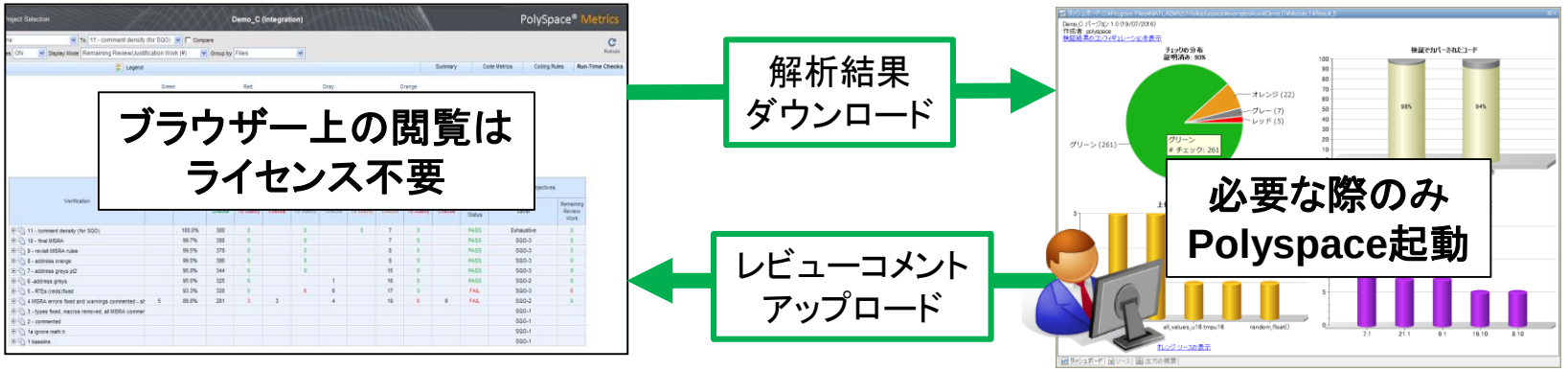
Polyspace Metrics：解析結果の一括管理・共有

メリット：統一レビュー容易化・ライセンス利用効率化

- 複数メンバーのレビュー作業を統一管理
- 面倒なレビュー情報の統合作業を削減
- Polyspace UIでの確認の必要性を明確化

Polyspace基本ツール機能による利便性向上

- プロジェクト品質の進捗状況を可視化
- ウェブブラウザ上で結果概要を確認
- 1つのデータベースに様々な開発者がコメントを記入



CI = Continuous Integration

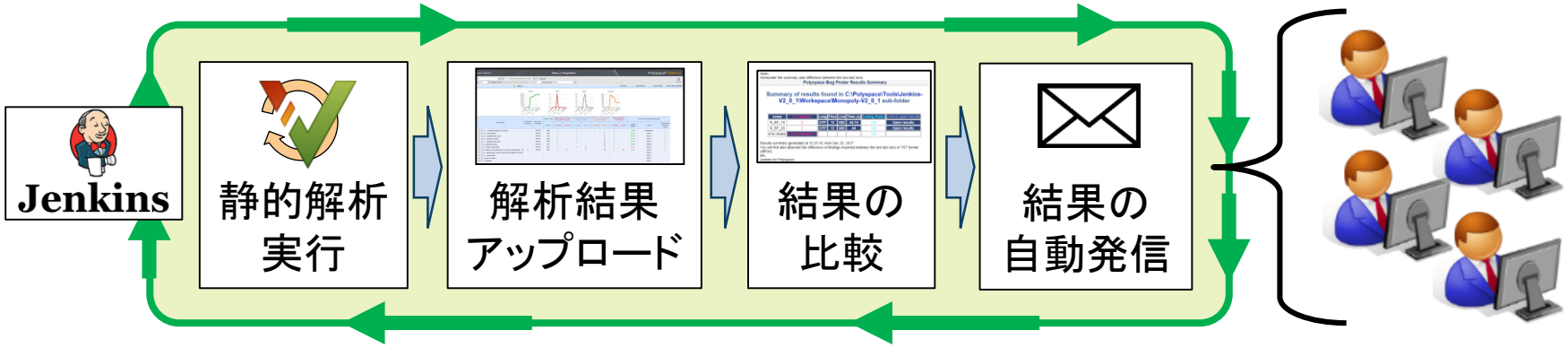
静的解析の自動化トレンド：CIツール連携

メリット：自動的な品質チェックと結果共有

- マニュアルの解析実行作業を削減
- 必要に応じた解析ジョブの実行（定期的、他タスク実行時）
- Polyspace Metricsへのアップロードによる結果概要共有

Polyspace解析実行スクリプトは自動作成可能

- CIツール連携にて下記ステップを自動化



デベロッパー向けのEclipse連携機能

メリット：デベロッパーの静的解析を加速化

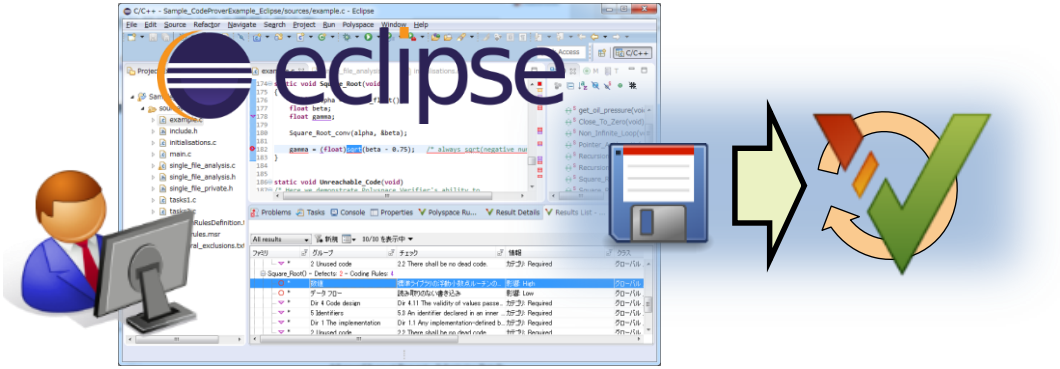
- PolyspaceのEclipseプラグイン利用
- Eclipse IDE内の活用による容易化

ソースコード保存時に解析を自動実行

- コード作成後に迅速なフィードバック
- 考えずに実行する定期的な解析

Polyspace Bug Finderの「高速解析モード」

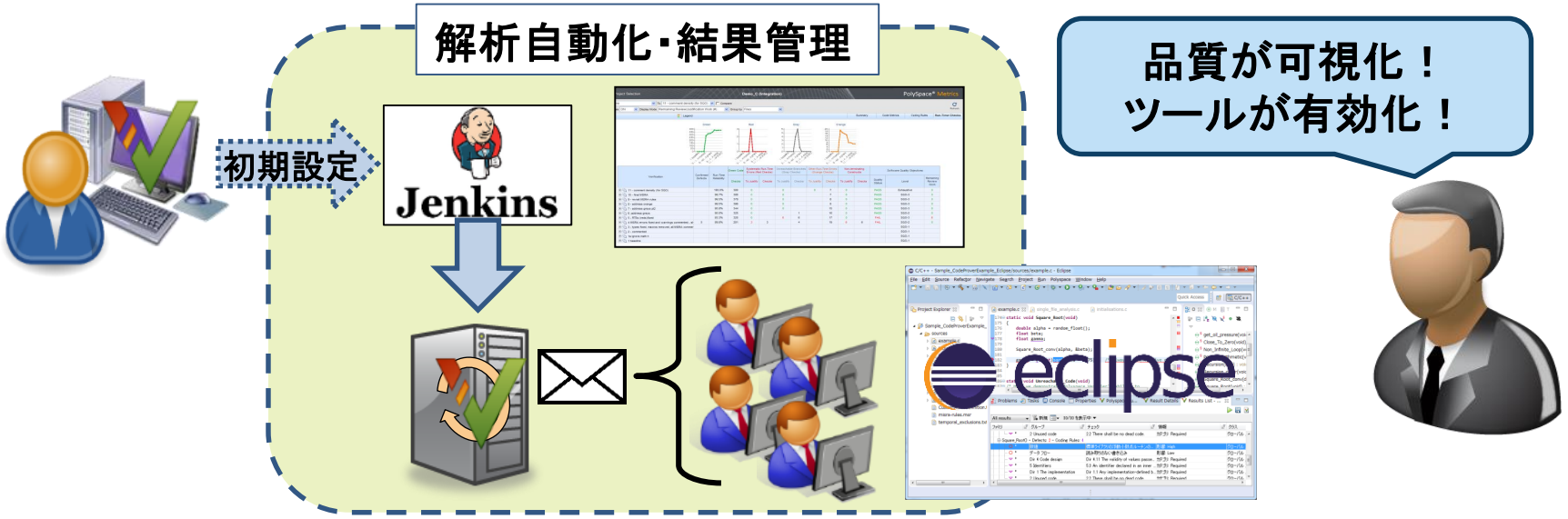
- 派生開発に有効な差分解析機能



まとめ：マニュアル作業を低減するプロセス効率化

静的解析の効果を最大限にするため、
有効機能の導入とプロセス自動化を推奨します

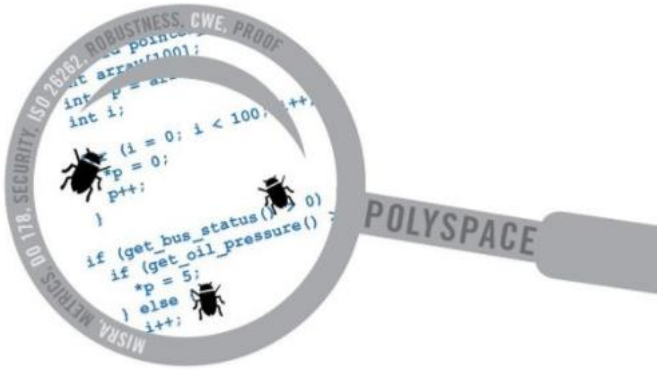
- Polyspaceは**パワフルな静的解析**の提供の上、便利機能により**チーム間レビューを推進**
- 静的解析プロセスを**自動化**することにより品質チェックを**容易化・定期化・効率化**

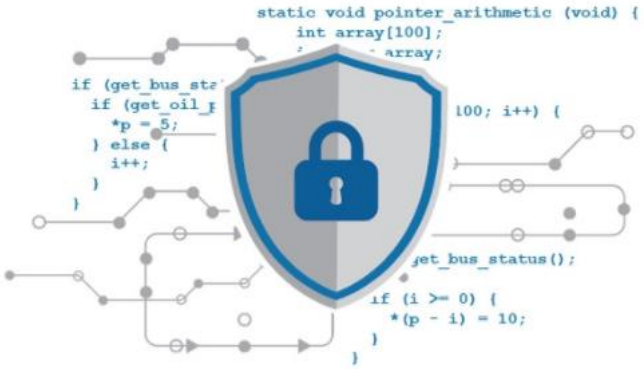


アジェンダ

- イントロダクション
- Polyspace静的解析の特徴
- 静的解析に関する課題と解決策

ユーザ事例 & 最後に. . .





Polyspace ユーザー事例
医療、航空、自動車など様々な分野で利用されています

Miracor Eliminates Run-Time Errors and Reduces Testing Time for Class III Medical Device Software

> "From a developer's perspective, the main advantage of Polyspace Code Prover is a higher level of quality and correctness in the code. Polyspace Code Prover helps Miracor demonstrate this quality and correctness to the regulatory community, including the FDA, to prove that our device is safe."
> *- Lars Schiemanck, Miracor Medical Systems*

https://jp.mathworks.com/company/user_stories/miracor-eliminates-run-time-errors-and-reduces-testing-time-for-class-iii-medical-device-software.html

Solar Impulse Develops Advanced Solar-Powered Airplane

> "From a developer's perspective, the main advantage of Polyspace Code Prover is a higher level of quality and correctness in the code. Polyspace Code Prover helps Miracor demonstrate this quality and correctness to the regulatory community, including the FDA, to prove that our device is safe."
> *- Lars Schiemanck, Miracor Medical Systems*

https://jp.mathworks.com/company/user_stories/solar-impulse-develops-advanced-solar-powered-airplane.html

日産自動車 ソフトウェアの信頼性を向上

> **「Polyspace 製品によって、高レベルなソフトウェアの信頼性を確保することができます。これは、業界内の他のツールにはできないことです。」**
> - 菊池光彦氏, 日産自動車

課題

ソフトウェア品質を向上させるために、発見が困難なランタイムエラーを特定する

ソリューション

MathWorks のPolyspace製品を使用して、日産とサプライヤーのコードを包括的に解析する

結果

サプライヤーのバグを検出して評価
ソフトウェアの信頼性が向上
日産のサプライヤーが Polyspace 製品を採用

https://jp.mathworks.com/company/user_stories/nissan-increases-software-reliability.html

最後に. . .

Polyspaceの評価にご興味が
ありましたらご連絡ください
（fred.noto@mathworks.co.jp）

Polyspace詳細にご興味が
ありましたら展示ブースにお立寄りください

Thank You For Your Attention
ご清聴ありがとうございました