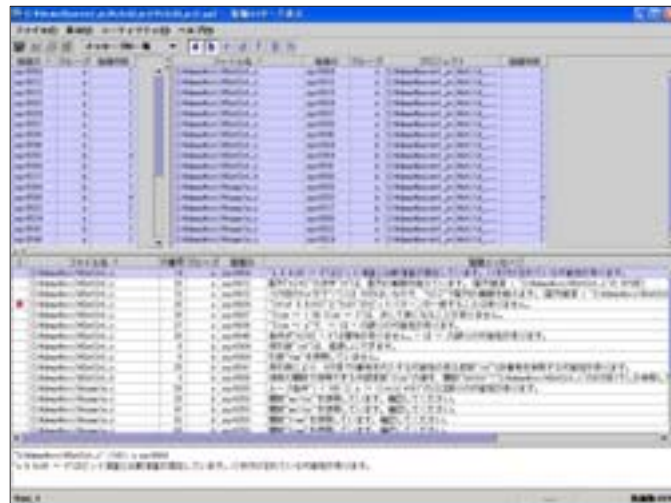


洗練された使いやすさのコア機能

解析エンジンが指摘した問題箇所を、効率よく的確に検証できます。

**機能1** 着目したい観点や関数の複雑さで解析できます。

PGReliefは、豊富な指摘種別を備え、また指摘種別フィルタリングにより、検証したい観点での問題箇所の絞り込みが容易にできます。また、複雑な関数の使用を抑えるための複雑さ計測機能も搭載しています。



**全体解析／差分解析／選択解析**

プロジェクトに登録されたソースコードに対し、すべてを解析する全体解析、解析する必要のあるものだけを解析する差分解析、選択したものだけを解析する選択解析があります。無駄な解析を避け、思い通りの検証作業が行えます。

**複雑さ計測**



関数ごとの複雑さを計測し、アラーム表示することで、バグの温床を未然に防止します。

**命名規約検査**

例：命名規約ファイル  
この命名規約ファイルが規定している内容  
①グローバル変数は、変数名をg\_で始めること  
PREFIX = g\_  
②グローバル変数で、かつuint型の変数は、g\_unで始めること  
uint = un

```
指摘例
1 typedef unsigned int uint;
2
3 uint g_size = 0;
4
5 void SetCntProc( uint n )
6 {
7
```

pgl1301グローバル変数" g\_Size"は命名規約 (uint = un) に違反しています。

変数や関数の名前付けに関し、プレフィックス文字、大文字／小文字や数字、アンダーバーの使用について、管理者が定めた標準規約に応じて検査し、指摘します。可読性を高め、信頼性を向上します。

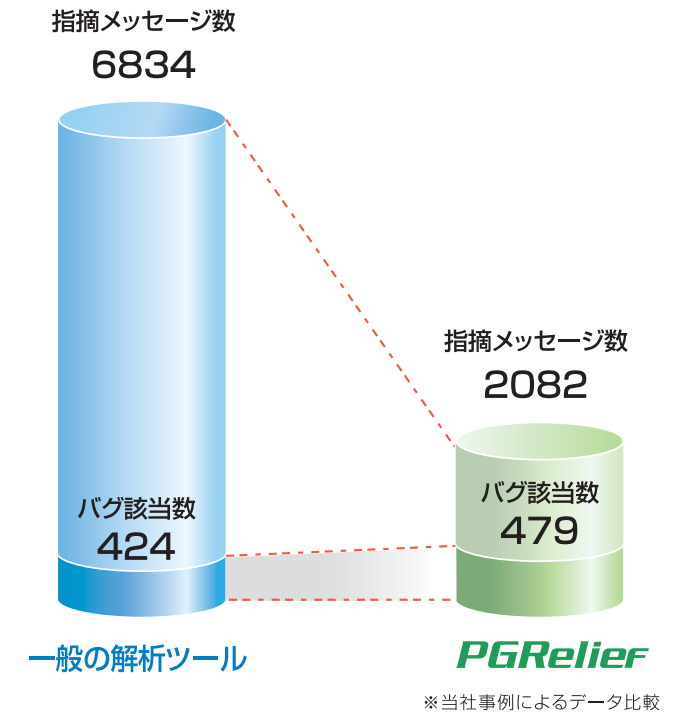
**群を抜く指摘数**

指摘種別	C指摘	C++指摘
障害の可能性	124個	94個
移植性・保守性・可読性の向上	221個	159個
移植性・保守性・可読性の向上 (とくにファームウェア開発)	96個	35個
堅牢性(セキュリティ)の向上	20個	20個
効率性の向上	5個	5個
命名規約違反	5個	5個
その他(構文の誤りなど)	42個	46個
計	513個	364個

**機能2** 精度の高い指摘で、ヒット率を向上させます。

プログラムの欠陥の可能性をピンポイントで高精度に指摘します。エラーの可能性を単純に指摘するのではなく、論理的な処理の流れを把握し、正当性を検証してエラーにならない場合は指摘しない仕組みを実現しています。ヒット率が高いと、指摘メッセージを確認する労力が軽減されます。

※ヒット率=バグ該当数/指摘メッセージ数



**機能3** 代表的コーディング作法に準拠し、適合性をチェックします。

SEC、MISRA-Cに基づいて記述されているかどうかをチェックし、違反している箇所を指摘します。



- ルール番号やルール分類で絞り込み表示できます。
- ビューアやエディタと連携し違反箇所に対処します。
- ルールのサブセットのみの指摘も可能です。
- 逸脱箇所のCSV形式での出力が可能です。
- 各ルールに対する指摘観点を表示します。

- SECルールは、129ルール中114ルールに対応、そのほかコーディングミスの例20個中18個についてもチェックします。
- MISRA-C V1は127ルール中120ルールに対応。
- MISRA-C V2は141ルール中131ルールに対応。

MISRA-C適合評価はオプションです