

# JUnitへのJava Pathfinderの適用

富士通株式会社

末松 慎也

suematsu.shinya@jp.fujitsu.com

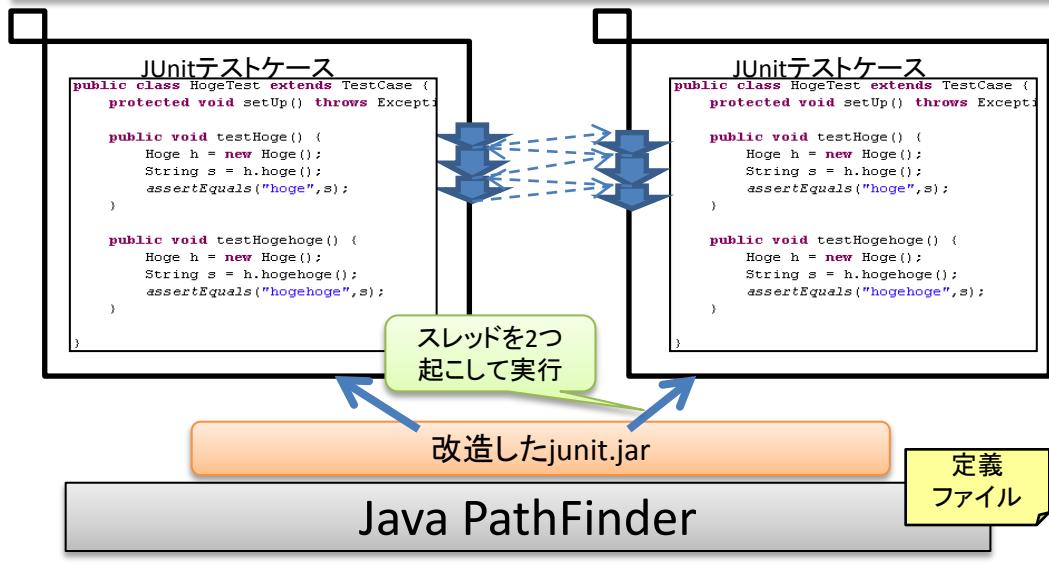
## 開発における問題点

オープンソースソフトウェアの品質保証としてJUnitテストが多く用いられているが、マルチスレッド起因のタイミングに依存したバグをJUnitで検出することは難しい。マルチスレッドでのバグを検出するJava Pathfinderというモデル検査ツールがあるが、使用法が難しい。

## 手法・ツールの連携による解決

JUnitテストケースをマルチスレッド実行できるようにJUnitライブラリを拡張し、さらにそれをJava Pathfinderと組合わせて実行できるようにした。ユニットテストとすることにより、Java Pathfinder使用時の壁である状態爆発を回避し、開発者は普段慣れているJUnitテストケースを作成すれば、容易にマルチスレッド検証を行える。

## 提案手法



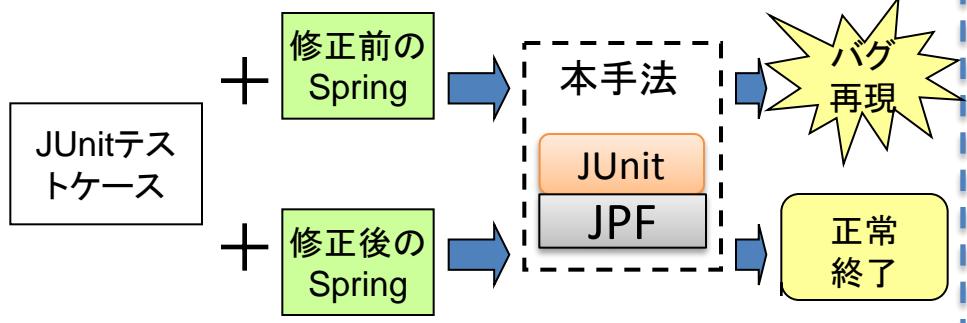
1. Java Pathfinderが直接的に実行するクラスにはJUnitライブラリのメインクラスを指定
2. JUnitのメインクラスへの引数としてJUnitテストケースクラスを指定
3. 拡張JUnitライブラリは、JUnitテストケースクラス内のテストメソッドをマルチスレッドで実行

マルチスレッドで例外が発生すればJava Pathfinderにより検出できる

## 適用事例

Spring Frameworkのバグによるトラブル

- タイミング依存であるため再現性が不安定
- 修正案がデグレードしていないか不明
- Java Pathfinderを使用しないとすると負荷テストで確認するしかない



本手法により確実な再現と修正妥当性の確認ができた。

## 評価

- テストケースをJUnitテストケースとして書けるため、JUnitに慣れた開発者に受け入れられやすい。  
→ 品質確保工程に無理なく取り込める
- シングルスレッドのJUnit実行では検出できないマルチスレッド起因バグを検出できることを確認。
- 実際の業務においてトラブル時の再現確認と修正の妥当性確認に活用できた。

## 今後の課題

Java Pathfinderを改善して下記の機能が使用できるようにする

- JUnit4
- easymock