



研究進捗状況

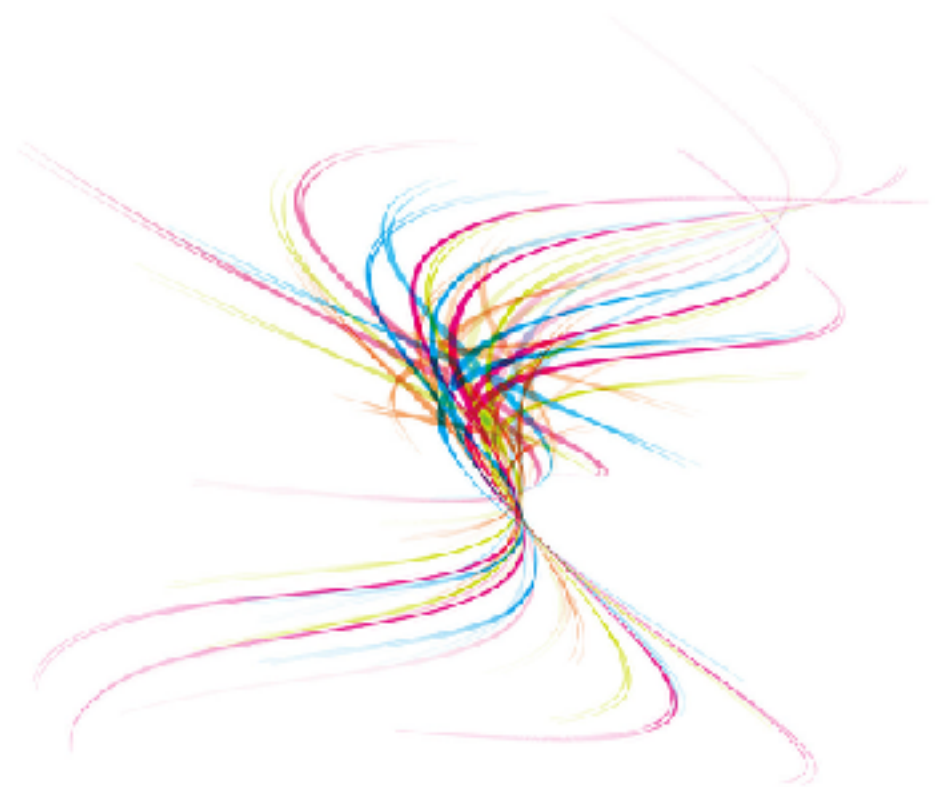
2018年0月0日（月）

発表者 吉富楓雅

目次

- 研究のイメージ
- 研究進捗状況
- 今後の展望・予定

研究のイメージ



研究タイトル(仮)

- 「プロダクトコードとテストコードの積集合の調査(?)
- 様々なオープンソースプロジェクトの開発が進むなか、多くのプロジェクトでは、大量のテストコードが開発中にプロダクトコードとともに進化している[1]
- プロダクトコードとテストコードの比が1:1とも言われている[2]
- つまり、プロダクトコードが開発されるにつれ、テストコードにもバグが混入する可能性が高まる

- [1] A. Zaidman, B. Van Rompaey, A. van Deursen, and S. Demeyer, “Studying the co-evolution of production and test code in open source and industrial developer test processes through repository mining,” Empirical Software Engineering, 16(3), pp. 325-364, June 2011.
- [2] A. van Deursen, L. Moonen, A. van den Bergh, and G. Kok, “Refactoring test code,” 2nd Int. Conf. on Extreme Programming and Flexible Processes in Software Engineering (XP 2001), 2001.

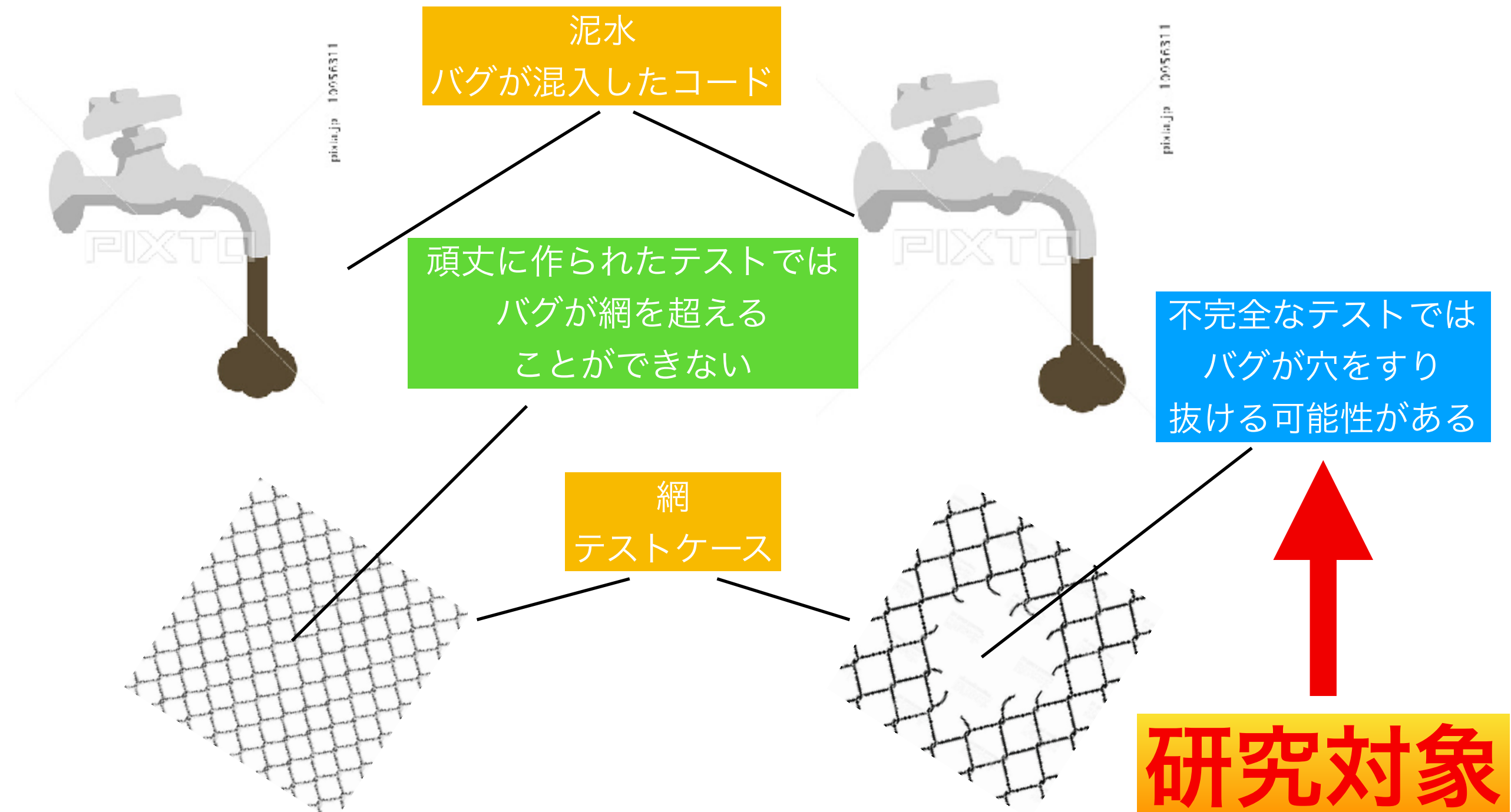
テストをテストする

- プロダクトコードに対して、テストコードは一般的に1～2倍の量を記述する
- このテストが適切であるかを調べるためには、この2倍の量のテストを行わなければならないのか
 - * 現場において、そのようなことは不可能
- では、テストのテストはどのようにすれば効率よく適切に行うことができるのか？

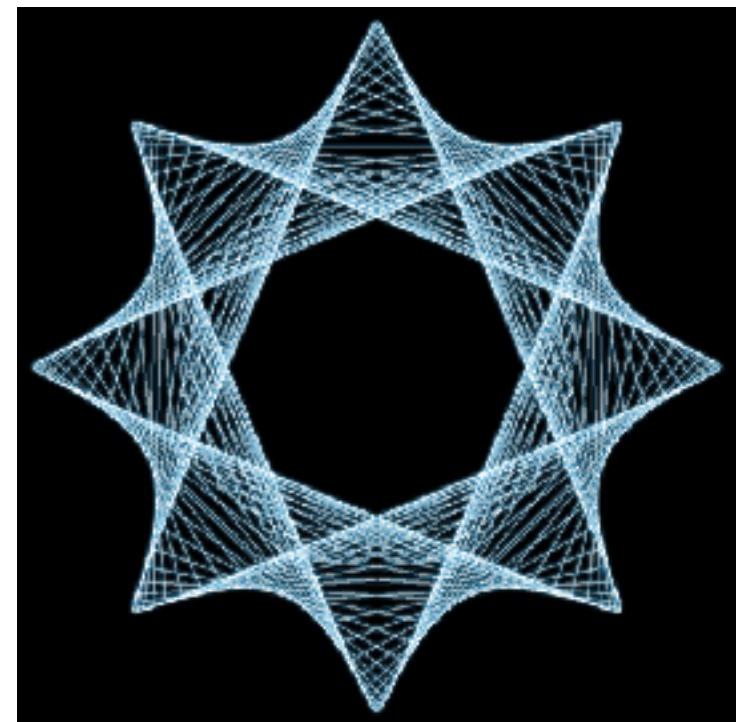
最悪のケース

- ソフトウェアテストにおいて 一番最悪なケースは, テストコードの誤りによってプロダクトコード内のバグが見逃されてしまうことである
 - * 開発者が頑丈なテストケースを作成したと思っても, 何かしらのミスでテストコードに綻びが生じていることは予想できる
 - * 柏先輩談: 実際に企業においても, プロダクトコードに対して書かれた膨大な量のテストコードを誰がテストするのかについて疑問を投げかける者がおり, またプログラマ自身も, 誤っている可能性があるコードに対して, テストコードが完全に網羅されているのか不安になりながらコード作成しなければならず, 心理的な負担も大きい.
- 本研究では, そのようなケースが(あるとは思うが)存在するのかを検証し, どのような事例の時に発生しやすいのかについて分析を行う.

イメージ



研究進捗状況



データのサンプリング

- 分析対象



- * Mybatis-3

- ▶ **MyBatis** :カスタム SQL, ストアドプロシージャ, 高度なマッピング処理に対応した永続化フレームワーク

- 150のテストデータをサンプリング
 - * 条件: バグIDに紐づけられており, プロダクトコード・テストコード共に修正あるいはコードの追加が見られるもの
- このうち, テストバグによってプロダクトコードのバグが見逃されているものを発見する

現状

- テンプレ

1. テンプレ

2. テンプレ

- テンプレ

3. テンプレ

4. テンプレ

今後の展望・予定



テンプレ

- テンプレ

- * テンプレ

- ▶ テンプレ

- テンプレ

- * テンプレ

- テンプレ

- * テンプレ

テンプレ

- テンプレ

1. テンプレ

2. テンプレ

- テンプレ

3. テンプレ

4. テンプレ