

# A Comparative Study of Bug Patterns in Java Cloned and Non-cloned Code

Hitesh Sajnani, Vaibhav Saini, and Cristina V. Lopes  
Bren School of Information and Computer Sciences  
University of California, Irvine, Irvine, CA, USA

片桐

# 背景

- コピー&ペーストによるソースコードの再利用は、一般的になっている
- コードクローンは有害であるとして考えられることが多かった
- 近年では違う見方も増えてきている

# 目的

- 新しい見方での研究が不足している
- コードクローンがソフトウェア品質に及ぼす影響に関する研究がほとんどない
- Survey of Research on Software Clonesの中に
  - ソフトウェア品質に対するコードクローンの関係

# FindBugs

- Javaプログラムの静的解析ツール
- 問題のあるコードを検出し、分類してくれる

The screenshot displays the FindBugs IDE interface. On the left, a tree view shows the hierarchy of bugs, with 'null とわかってる値の冗長な null チェック (2)' selected. The main window shows the source code of 'ObservableFlatMap.java' with a yellow highlight on line 352: `if (o == null) { break; }`. The bottom panel shows the bug details: 'null とわかってる値の冗長な null チェック' and 'Bug kind and pattern: RCN - RCN\_REDUNDANT\_NULLCHECK\_OF\_NULL\_VALUE'. The URL 'http://findbugs.sourceforge.net' is visible at the bottom left, and the University of Maryland logo is at the bottom right.

Class name filter:  Filter

Group bugs by: カテゴリー バグ種類 バグ-パターン Bug Rank 名称

バグ (896)

- Correctness (5)
- Bad practice (282)
- Malicious code vulnerability (25)
- Multithreaded correctness (23)
- Performance (150)
- Dodgy code (411)
- null への冗長な比較 (2)
- null とわかってる値の冗長な null チェック (2)
- null とわかってる値 o の冗長な null チェックがあります。drainLoop()
- null とわかってる値 o の冗長な null チェックがあります。drainLoop()
- null 値の利用 (1)
- 冗長なインタフェース (6)
- 書き込まれないフィールド (402)

ObservableFlatMap.java in io.reactivex.internal.operators.observable

```
336     if (svq != null) {
337         for (;;) {
338             U o;
339             for (;;) {
340                 if (checkTerminate()) {
341                     return;
342                 }
343             }
344             o = svq.poll();
345             if (o == null) {
346                 break;
347             }
348             child.onNext(o);
349         }
350     }
351     if (o == null) {
352         break;
353     }
354 }
355 }
356 }
357 }
358 }
359 }
360 }
361 }
362 }
363 }
364 }
365 }
366 }
367 }
368 }
```

検索 次を検索 前を検索

null とわかってる値 o の冗長な null チェックがあります。drainLoop()  
冗長な null チェックが行われている箇所 ObservableFlatMap.java:[line 352]  
メソッド io.reactivex.internal.operators.observable.ObservableFlatMap\$  
o からロードされた値  
戻り値 io.reactivex.internal.fuseable.SimplePlainQueue.poll()

**null とわかってる値の冗長な null チェック**  
このメソッドには null とわかってる値の冗長な null チェックがあります。

Bug kind and pattern: RCN - RCN\_REDUNDANT\_NULLCHECK\_OF\_NULL\_VALUE

<http://findbugs.sourceforge.net>

UNIVERSITY OF MARYLAND

# FindBugs カテゴリ(一部)

| カテゴリ名                      | 概要                                                               |
|----------------------------|------------------------------------------------------------------|
| Correctness                | バグの可能性 - 明らかなコーディングミスによって、開発者の意図しないコードになっている。誤検出は最小限に留めなければならない。 |
| Multi-threaded correctness | スレッド、ロック、volatileの扱いに対するコードの欠陥。                                  |
| Performance                | 必ずしも間違いではないが、実行効率が悪くなる可能性のあるもの。                                  |
| Security                   | リモートから悪用可能なセキュリティ上の脆弱性に繋がる、信頼されていない入力方法。                         |

# FindBugs バグパターン

- 検出したバグがどのような誤りであるかを示す
  - URF\_UNREAD\_FIELD
    - Performanceカテゴリ
    - 読み出されないフィールド
  - NP\_NULL\_ON\_SOME\_PATH
    - Correctnessカテゴリ
    - null 値を利用している可能性がある

など

たくさんある

# RQ

1. クローンコードの欠陥密度は、クローンでないコードの密度より大きいのか？
  - 欠陥密度 = バグパターンの数 / 1,000(行数)
2. クローンコードで頻繁に見られる特定のバグカテゴリがあるか？
3. どのくらいの頻度でバグパターンがクローンによって伝播するか

# 分析手法

- クローン検出ツールの使用
  - メソッド単位でクローンコード・非クローンコードを検出
- FindBugsを適用
  - プロジェクトに存在する全てのバグパターンを検出
  - メソッドとバグパターンの関連付け
  - メソッドとバグカテゴリの関連付け
  - カテゴリの分解
    - プライマリカテゴリ(Correctness, Performance, Security等)
    - セカンダリカテゴリ(Bad Practice, Style等 影響が小さい)

# なぜFindBugsを使うのか

- ソフトウェアの品質の測定には、欠陥情報を扱うため、BTSや静的解析ツールが使用される。
- 今回は静的解析ツールのFindBugsが使用される
  - 1.BTSと、ソースコードリポジトリが対応しているとは限らない
  - 2.BTSに報告される不具合が誤っていることがある
  - 3.BTSに報告される不具合数はプロジェクトの人気に相関している
  - 4.プロジェクトの存続期間に応じて報告される不具合数に影響する

# 対象プロジェクト

- ApacheのJavaプロジェクト 31プロジェクト

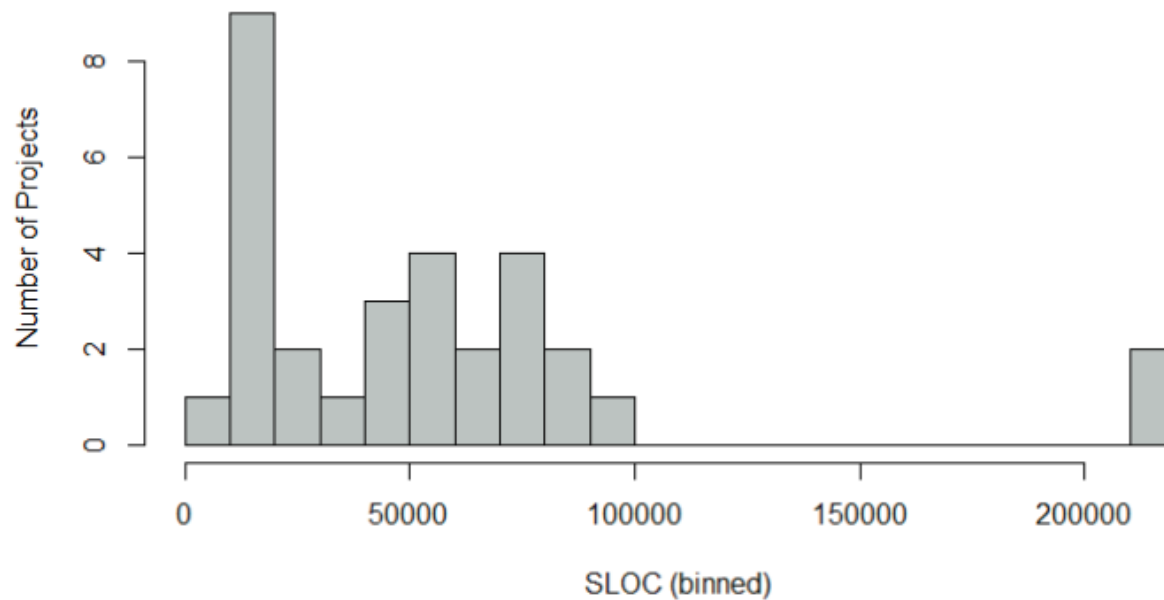


Fig. 1: Size distribution of projects. Size is defined as non-commented lines of code

# 分析結果 RQ1

RQ1 クローンコードの欠陥密度は、クローンでないコードの密度より大きいのか？

- 31プロジェクト中、26プロジェクトがクローンコードの欠陥密度がクローンでないコードよりも低い

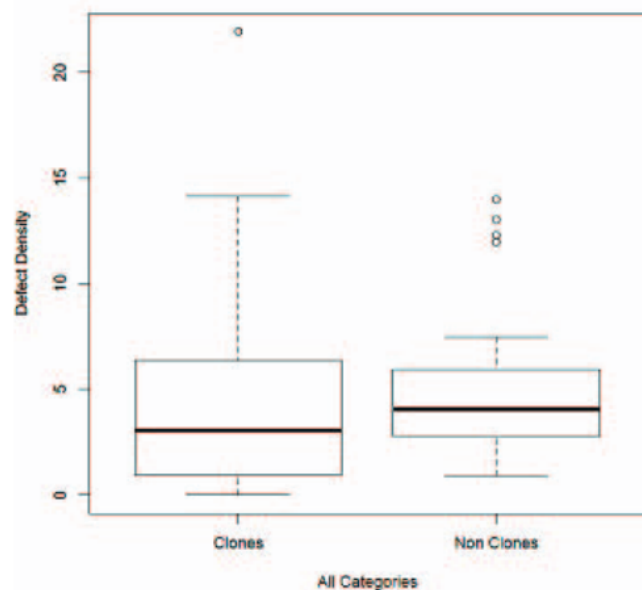
| Project         | SLOC    |        | #Method |        | Defect Density |        | Correlation | Defect Density (Non-clones)<br>(size control) |           | Bugs      |                |
|-----------------|---------|--------|---------|--------|----------------|--------|-------------|-----------------------------------------------|-----------|-----------|----------------|
|                 | Total   | Clones | Total   | Clones | Non-clones     | Clones |             | Equal # Methods                               | Equal LoC | uplicated | non-duplicated |
| ant             | 86,438  | 3,642  | 7,883   | 208    | 2.66           | 1.65   | 0.195       | 2.46                                          | 2.61      | 4         | 0              |
| berkeleyparser  | 57,905  | 7,842  | 3,364   | 357    | 4.35           | 0.89   | 0.168       | 4.08                                          | 4.23      | 9         | 10             |
| cglib           | 13,668  | 354    | 1,916   | 28     | 2.69           | 0      | 0.123       | 3.74                                          | 1.65      | 0         | 0              |
| cloud9          | 56,766  | 21,066 | 4,390   | 1,047  | 11.96          | 6.5    | 0.149       | 5.02                                          | 5.07      | –         | –              |
| cocoon          | 10,387  | 211    | 773     | 25     | 2.08           | 4.92   | 0.197       | 3.87                                          | 1.82      | 1         | 0              |
| commons-io      | 8,673   | 440    | 674     | 50     | 1.05           | 0      | 0.137       | 2.97                                          | 3.56      | 1         | 1              |
| dom4j           | 17,854  | 997    | 2,206   | 72     | 3.74           | 0      | 0.096       | 0.82                                          | 0.8       | 1         | 0              |
| hadoop-hdfs     | 70,411  | 5,016  | 5,401   | 362    | 2.48           | 2.03   | 0.051       | 1.05                                          | 1         | 0         | 1              |
| hadoop-mapred   | 64,023  | 3,452  | 5,577   | 230    | 3.05           | 1.28   | 0.205       | 2.4                                           | 2.25      | 2         | 1              |
| hibernate       | 72,409  | 3,117  | 8,973   | 243    | 4.18           | 0      | 0.017       | 2.79                                          | 2.97      | 6         | 1              |
| httpclient      | 18,022  | 1,206  | 1,324   | 102    | 3.06           | 6.8    | 0.101       | 1.6                                           | 1.81      | 2         | 0              |
| jfreechart      | 93,460  | 15,470 | 7,034   | 871    | 1.01           | 0.2    | 0.214       | 0.89                                          | 0.72      | 89        | 31             |
| jython          | 211,905 | 7,411  | 19,593  | 620    | 1.47           | 1.66   | 0           | 0.1                                           | 0.22      | 1         | 2              |
| log4j           | 16,111  | 609    | 1,687   | 57     | 0.86           | 0.58   | 0.142       | 2.32                                          | 2.24      | 2         | 1              |
| lucene          | 15,670  | 1,438  | 1,315   | 77     | 0.14           | 0.13   | 0.152       | 1.66                                          | 1.06      | 4         | 2              |
| mahout-core     | 53,366  | 3,800  | 3,943   | 249    | 1.61           | 0      | 0.003       | 1.54                                          | 1.09      | 0         | 0              |
| mason           | 35,931  | 3,800  | 2,674   | 261    | 1.48           | 0      | 0.278       | 1.37                                          | 1.36      | 8         | 5              |
| nutch           | 12,243  | 519    | 861     | 40     | 4.24           | 0      | 0.299       | 4.49                                          | 4.34      | 1         | 0              |
| pdfbox          | 13,936  | 1,653  | 894     | 31     | 12.27          | 0      | 0.256       | 14.81                                         | 9.74      | 0         | 0              |
| pig             | 84,770  | 3,059  | 5,458   | 180    | 3.03           | 0.33   | 0.157       | 2.97                                          | 2.27      | 1         | 7              |
| pmd             | 60,060  | 6,864  | 4,634   | 254    | 0.79           | 0.58   | 0.33        | 0.67                                          | 0.7       | 76        | 2              |
| poi             | 47,804  | 5,093  | 4,899   | 343    | 4.87           | 2.16   | 0.117       | 4.8                                           | 4.18      | 6         | 9              |
| postgresql      | 23,514  | 1,916  | 1,836   | 160    | 10.22          | 21.91  | 0.415       | 8.64                                          | 8.07      | 20        | 2              |
| rhino           | 54,722  | 2,412  | 2,958   | 135    | 0.87           | 0      | 0.209       | 0.5                                           | 2.4       | 0         | 0              |
| stanford-nlp    | 210,233 | 4,430  | 8,692   | 295    | 5.72           | 9.26   | 0.192       | 3.18                                          | 3.45      | –         | –              |
| struts          | 24,799  | 1,275  | 2,300   | 82     | 1.61           | 0.78   | 0.127       | 1.87                                          | 1.24      | 0         | 1              |
| substance       | 47,361  | 5,014  | 3,026   | 213    | 1.32           | 0.4    | 0.233       | 1.13                                          | 0.84      | 8         | 1              |
| synapse-core    | 41,612  | 2,370  | 2,803   | 138    | 0.78           | 0      | 0.18        | 0.88                                          | 1         | 1         | 2              |
| tomcat-catalina | 73,673  | 4,030  | 4,757   | 232    | 2.32           | 0.99   | 0.171       | 2.26                                          | 2.76      | 5         | 2              |
| uima-core       | 11,942  | 687    | 947     | 59     | 2.14           | 0      | 0.007       | 2.53                                          | 0.8       | 0         | 0              |
| xerces          | 76,185  | 7,340  | 5,131   | 345    | 0.91           | 0.68   | 0.205       | 1                                             | 0.87      | 6         | 4              |

TABLE I: Results. Correlation column shows Pearson correlation coefficient between method size and # of bug patterns. Defect density (size control) shows defect density of non-cloned code when (i) using the same number of non-cloned methods as cloned code methods (Equal # Methods); and (ii) using non-cloned methods whose LoC sums up to total cloned method LoC (Equal LoC)

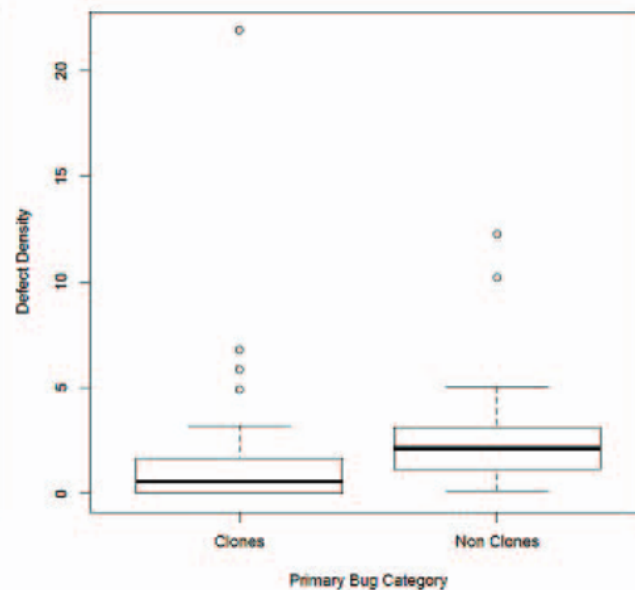
# 分析結果 RQ1

RQ1 クローンコードの欠陥密度は、クローンでないコードの密度より大きいのか？

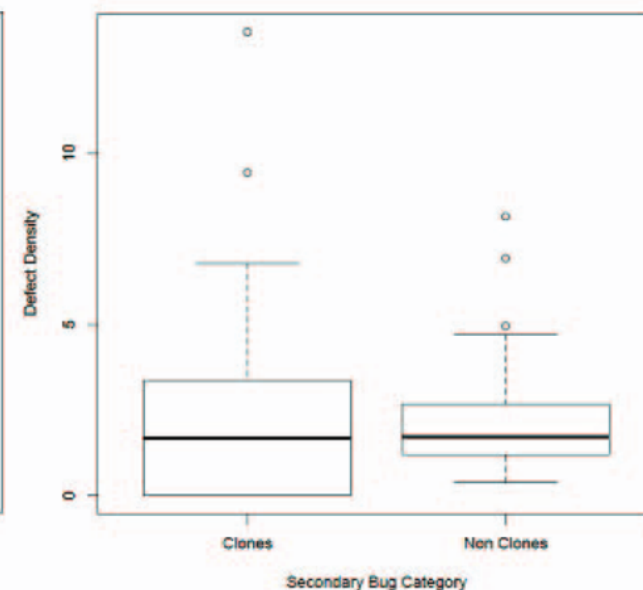
- クローンでないコードの方がクローンコードよりも欠陥密度が27位大きい、 $\rightarrow$  RQ1はNO



中央値の差：1.0



中央値の差：1.56



中央値の差：0.02

# 分析結果 RQ2

## RQ2

クローンコードで頻繁に見られる特定のバグカテゴリがあるか？

- クローンコードで検出されたバグパターンのうち、66%(606件中、399件)がStyleとBad Practiceカテゴリ。  
クローンでないコードでは49%(4206件中2065件)であった。
- RQ2 →

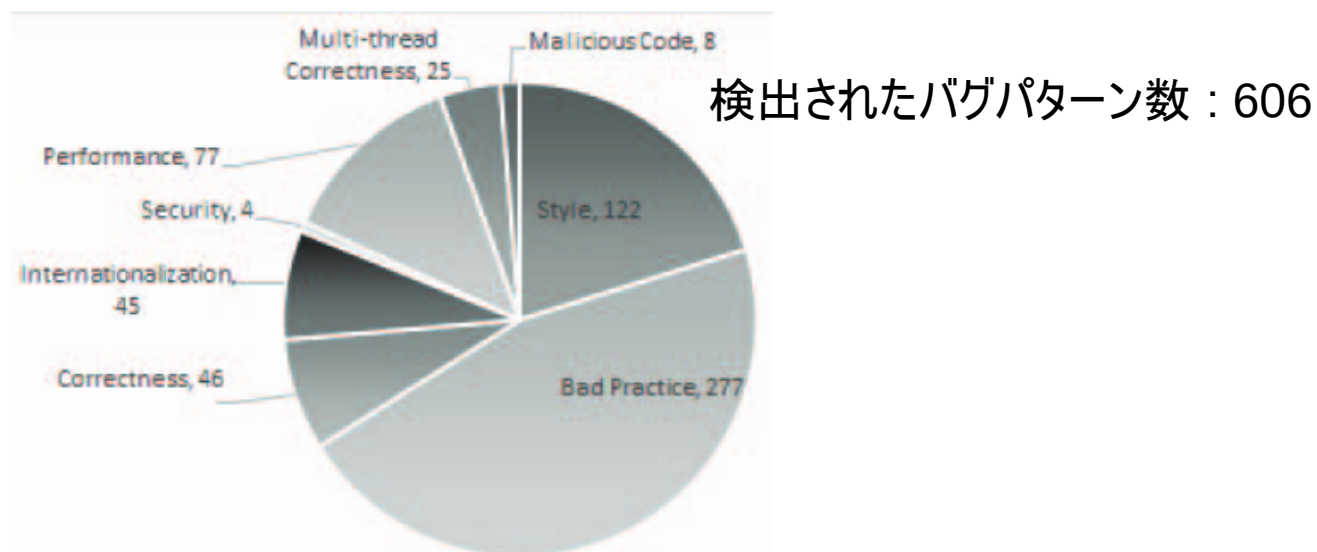


Fig. 4: Bug patterns in cloned code classified into various categories

# 分析結果 RQ3

## RQ3

どのくらいの頻度でバグパターンがクローンによって伝播するか

- バグパターンを含む339個のコードクローングループを目視で調べ、グループを2つに分類する
  1. バグパターンがグループ内で重複しているグループ (75%)
  2. バグパターンがグループ内で独立しているグループ (25%)
- クローンコードのバグパターンの75%は複製されていた

# 結論

- コードクローンとFindBugsに報告されるバグパターンの関係を31のプロジェクトを分析して調べた
- コードクローンが必ずしも有害ではないということが確認できた

# 所感

- FindBugsを使って、どのようにクローンと関連付けさせているのかわかった