Patched Clones and Missed Patches among Variants of a Software Family

John Businge

Assistant Professor – UNLV

Never Work in Theory (NWiT) – Spring 2023

Context

$425M

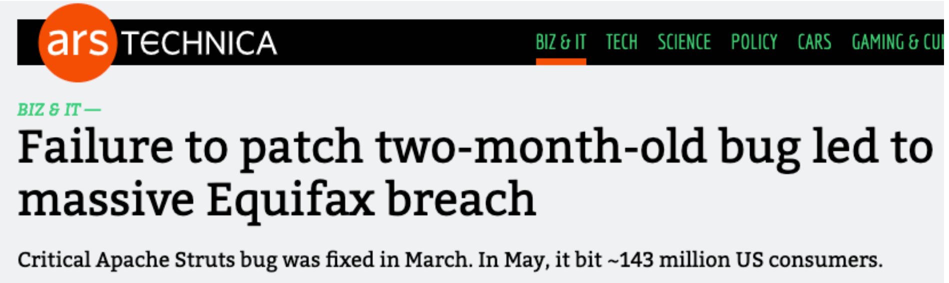
The Equifax data breach occurred between **May and July 2017** at the American credit bureau Equifax. Private records of 147.9 million Americans along with 15.2 million British citizens and about 19,000 Canadian citizens were compromised in the breach, making it one of the largest cybercrimes related to identity theft.

Wired Magazine, “Equifax has no excuse”, September 2017

open source

March 2017

CVE-2017- 5638



BIZ & IT —

Failure to patch two-month-old bug led to massive Equifax breach

Critical Apache Struts bug was fixed in March. In May, it bit ~143 million US consumers.

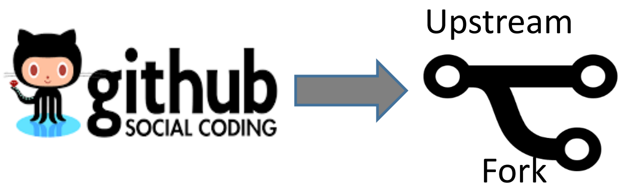
Recommender system

EQUIFAX DATA BREACH

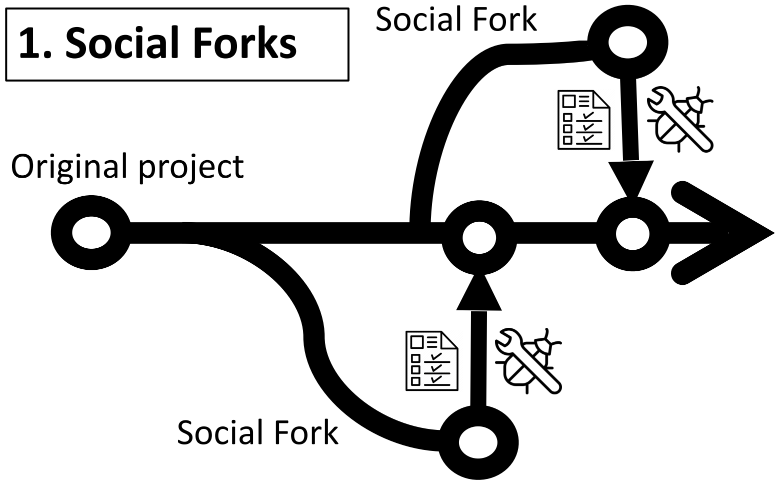
May 2017

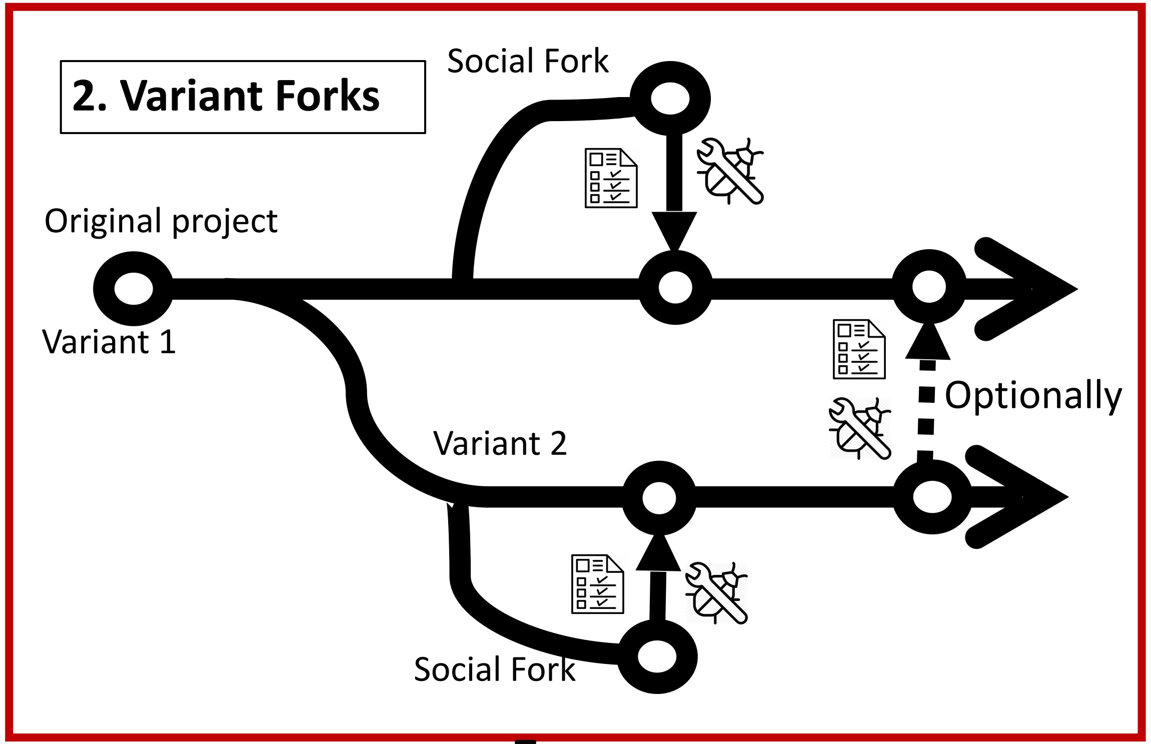
https://www.istockphoto.com/

Patched Clones and Missed Patches among Variants of a Software Family

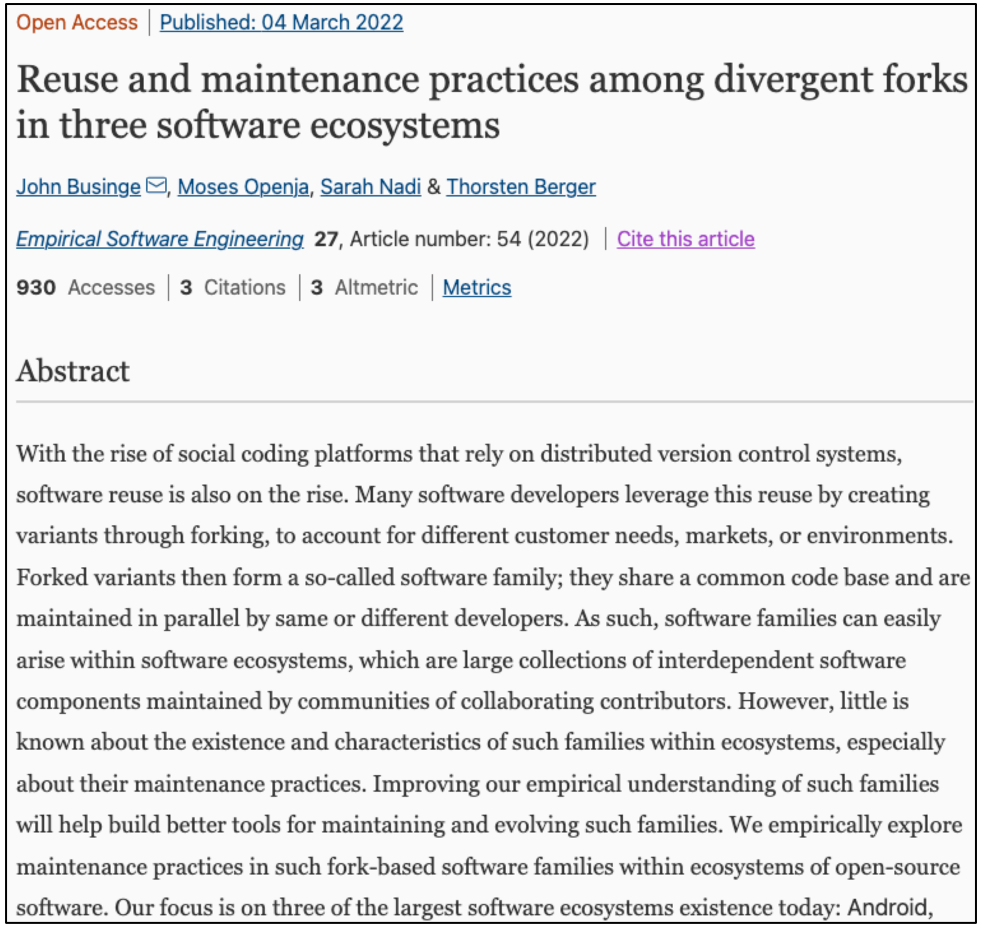


Software family – >= 2 variants


1. Social Forks


2. Variant Forks

Focus



Open Access | Published: 04 March 2022

Reuse and maintenance practices among divergent forks in three software ecosystems

John Businge, Moses Openja, Sarah Nadi & Thorsten Berger

Empirical Software Engineering 27, Article number: 54 (2022) | Cite this article

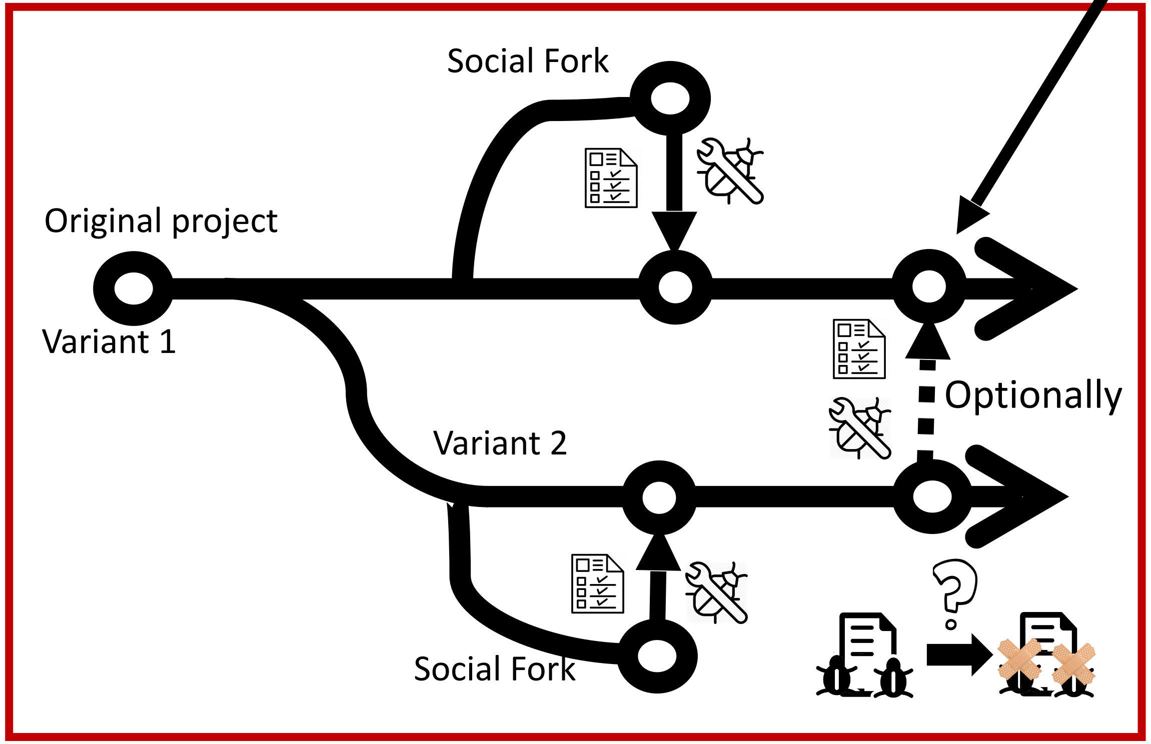
930 Accesses | 3 Citations | 3 Altmetric | Metrics

Abstract

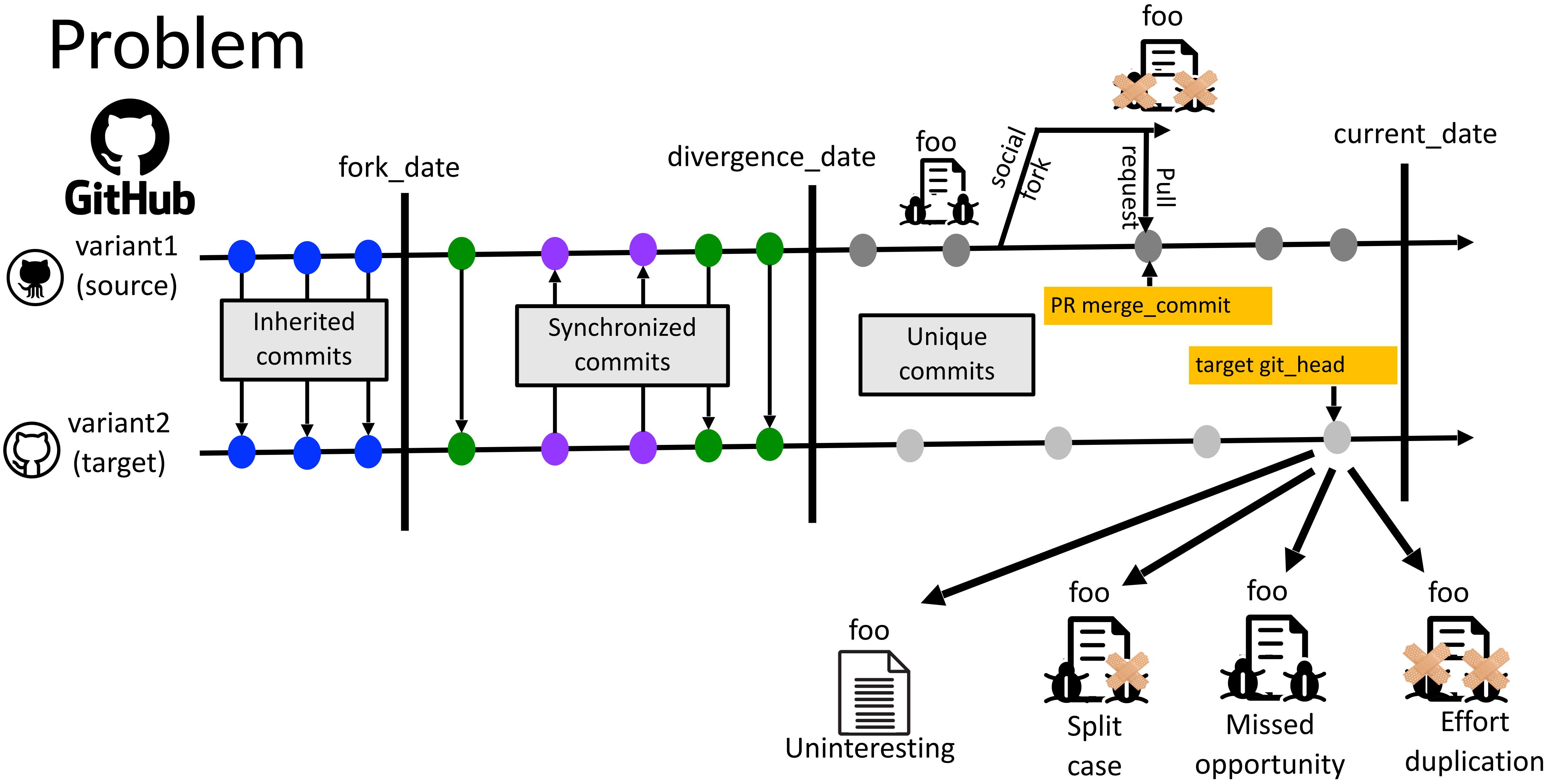
With the rise of social coding platforms that rely on distributed version control systems, software reuse is also on the rise. Many software developers leverage this reuse by creating variants through forking, to account for different customer needs, markets, or environments. Forked variants then form a so-called software family; they share a common code base and are maintained in parallel by same or different developers. As such, software families can easily arise within software ecosystems, which are large collections of interdependent software components maintained by communities of collaborating contributors. However, little is known about the existence and characteristics of such families within ecosystems, especially about their maintenance practices. Improving our empirical understanding of such families will help build better tools for maintaining and evolving such families. We empirically explore maintenance practices in such fork-based software families within ecosystems of open-source software. Our focus is on three of the largest software ecosystems existence today: Android,

>10K variants from three ecosystems on Github

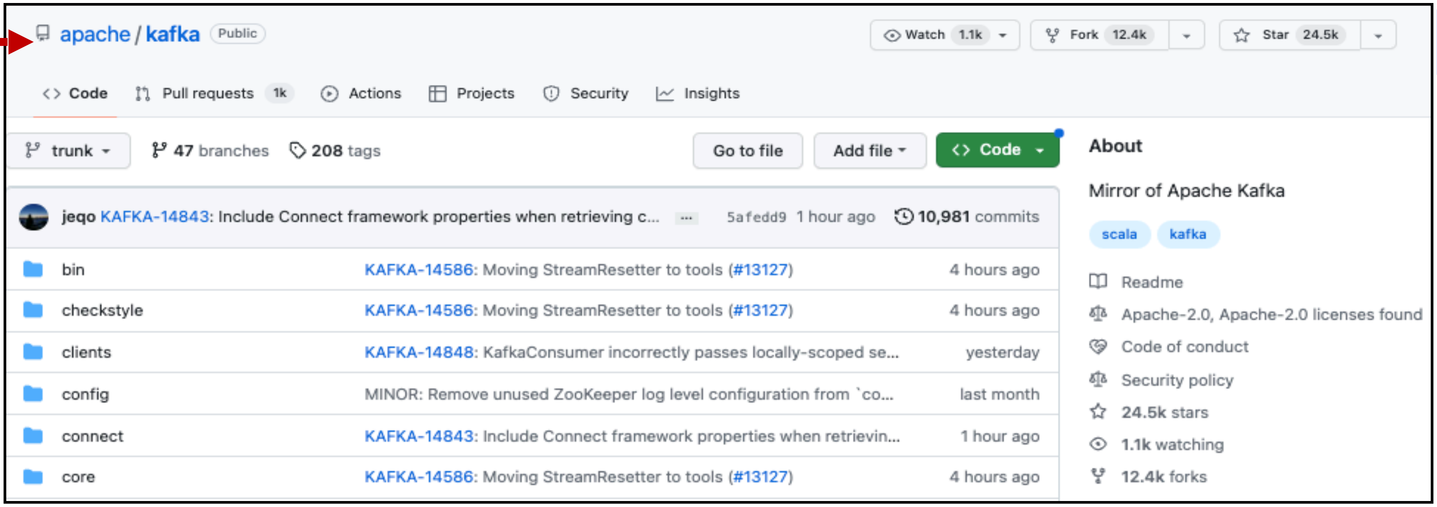
Rarely share updates



Problem


GitHub
variant1
(source)
variant2
(target)
fork_date
divergence_date
current_date
Inherited commits
Synchronized commits
Unique commits
foo
social fork
Pull request
foo
PR merge_commit
target git_head
foo
Uninteresting
foo
Split case
foo
Missed opportunity
foo
Effort duplication

Concrete Example



apache / kafka Public

47 branches 208 tags

jeqo KAFKA-14843: Include Connect framework properties when retrieving c... 5afedd9 1 hour ago 10,981 commits

bin	KAFKA-14586: Moving StreamResetter to tools (#13127)	4 hours ago
checkstyle	KAFKA-14586: Moving StreamResetter to tools (#13127)	4 hours ago
clients	KAFKA-14848: KafkaConsumer incorrectly passes locally-scoped se...	yesterday
config	MINOR: Remove unused ZooKeeper log level configuration from `co...	last month
connect	KAFKA-14843: Include Connect framework properties when retrievin...	1 hour ago
core	KAFKA-14586: Moving StreamResetter to tools (#13127)	4 hours ago

About

Mirror of Apache Kafka

scala kafka

Readme

Apache-2.0, Apache-2.0 licenses found

Code of conduct

Security policy

24.5k stars

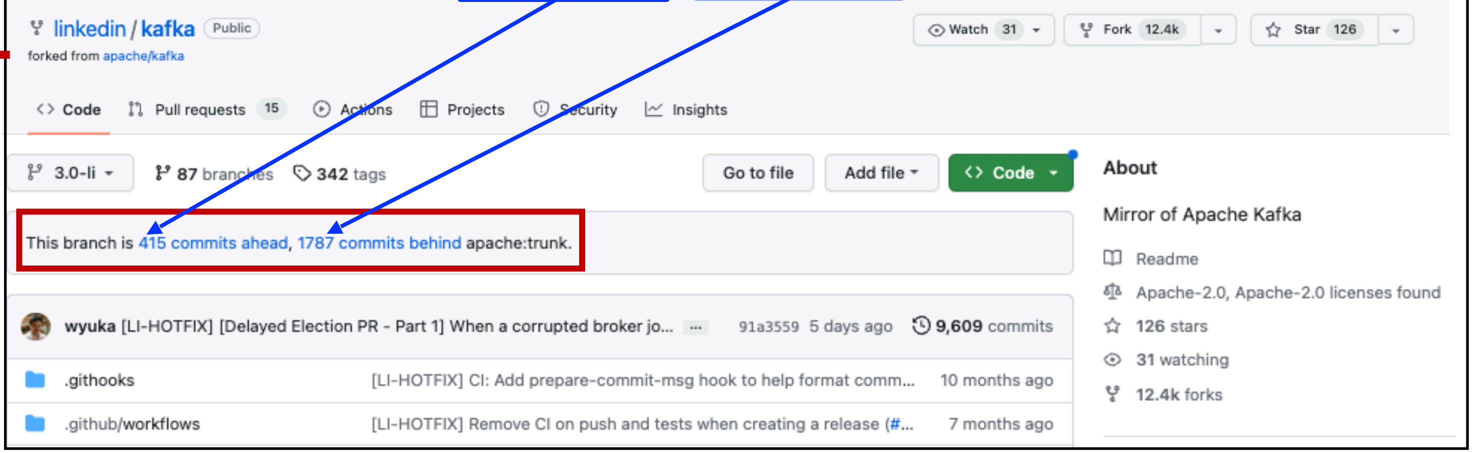
1.1k watching

12.4k forks

extraction_date 2023-03-27

415 linkedin unique commits

1,787 apache unique commits



linkedin / kafka Public

forked from apache/kafka

87 branches 342 tags

This branch is 415 commits ahead, 1787 commits behind apache:trunk.

wyuka [LI-HOTFIX] [Delayed Election PR - Part 1] When a corrupted broker jo... 91a3559 5 days ago 9,609 commits

.githooks	[LI-HOTFIX] CI: Add prepare-commit-msg hook to help format comm...	10 months ago
.github/workflows	[LI-HOTFIX] Remove CI on push and tests when creating a release (#...	7 months ago

About

Mirror of Apache Kafka

Readme

Apache-2.0, Apache-2.0 licenses found

126 stars

31 watching

12.4k forks

This is the version of Kafka running at LinkedIn.

Concrete Example:

Missed Opportunity

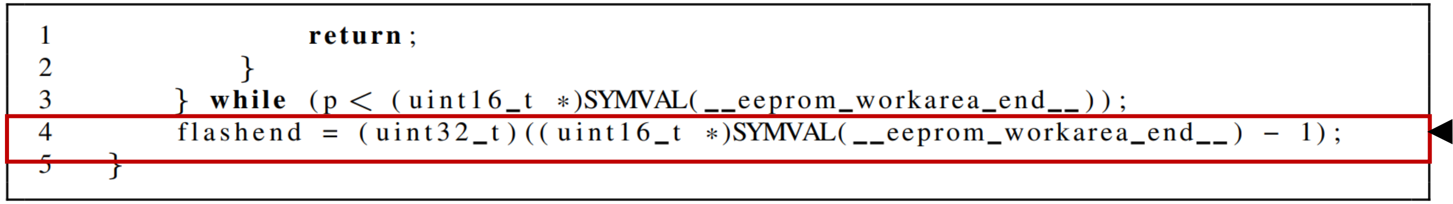
1 file - Pull request

extraction_date 2021-07-20

gcc10 [...] build warning #12587

Buggy code from upstream

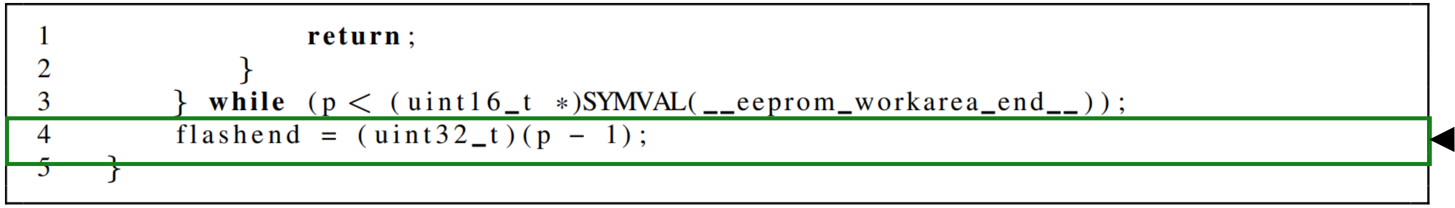
qmk/qmk_firmware



```
            return;
        }
    } while (p < (uint16_t *)SYMVAL(__eeprom_workarea_end__));
    flashend = (uint32_t)((uint16_t *)SYMVAL(__eeprom_workarea_end__) - 1);
}
```

Buggy line

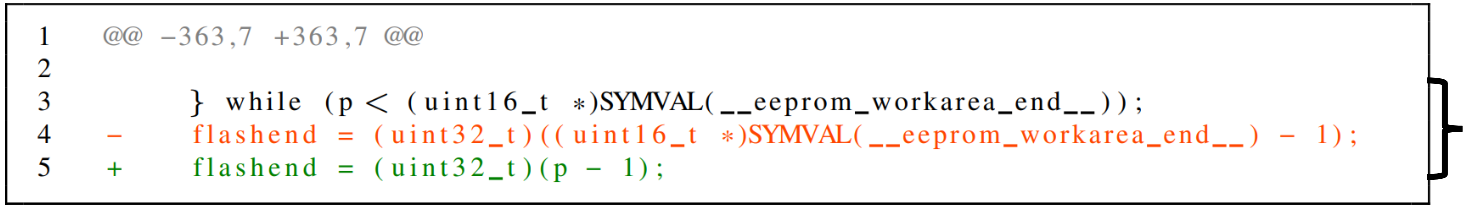
Patched code from upstream (Pull request)



```
            return;
        }
    } while (p < (uint16_t *)SYMVAL(__eeprom_workarea_end__));
    flashend = (uint32_t)(p - 1);
}
```

Patched line

Diff for patch in upstream



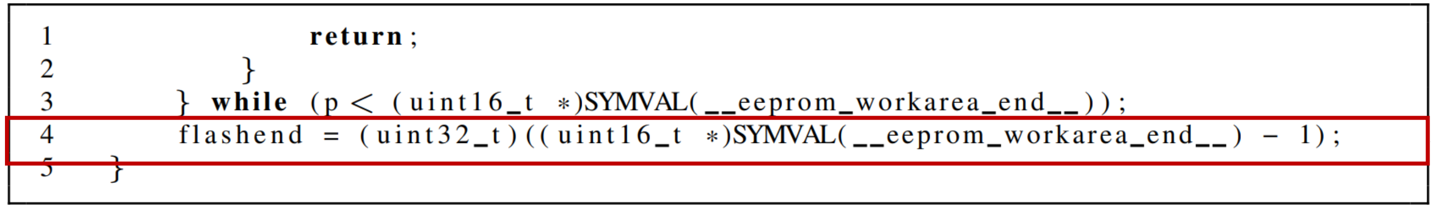
```
@@ -363,7 +363,7 @@

     } while (p < (uint16_t *)SYMVAL(__eeprom_workarea_end__));
-    flashend = (uint32_t)((uint16_t *)SYMVAL(__eeprom_workarea_end__) - 1);
+    flashend = (uint32_t)(p - 1);
```

Hunk

File from divergent fork at `git_head`

sekigon-gonnoc/qmk_firmware



```
            return;
        }
    } while (p < (uint16_t *)SYMVAL(__eeprom_workarea_end__));
    flashend = (uint32_t)((uint16_t *)SYMVAL(__eeprom_workarea_end__) - 1);
}
```

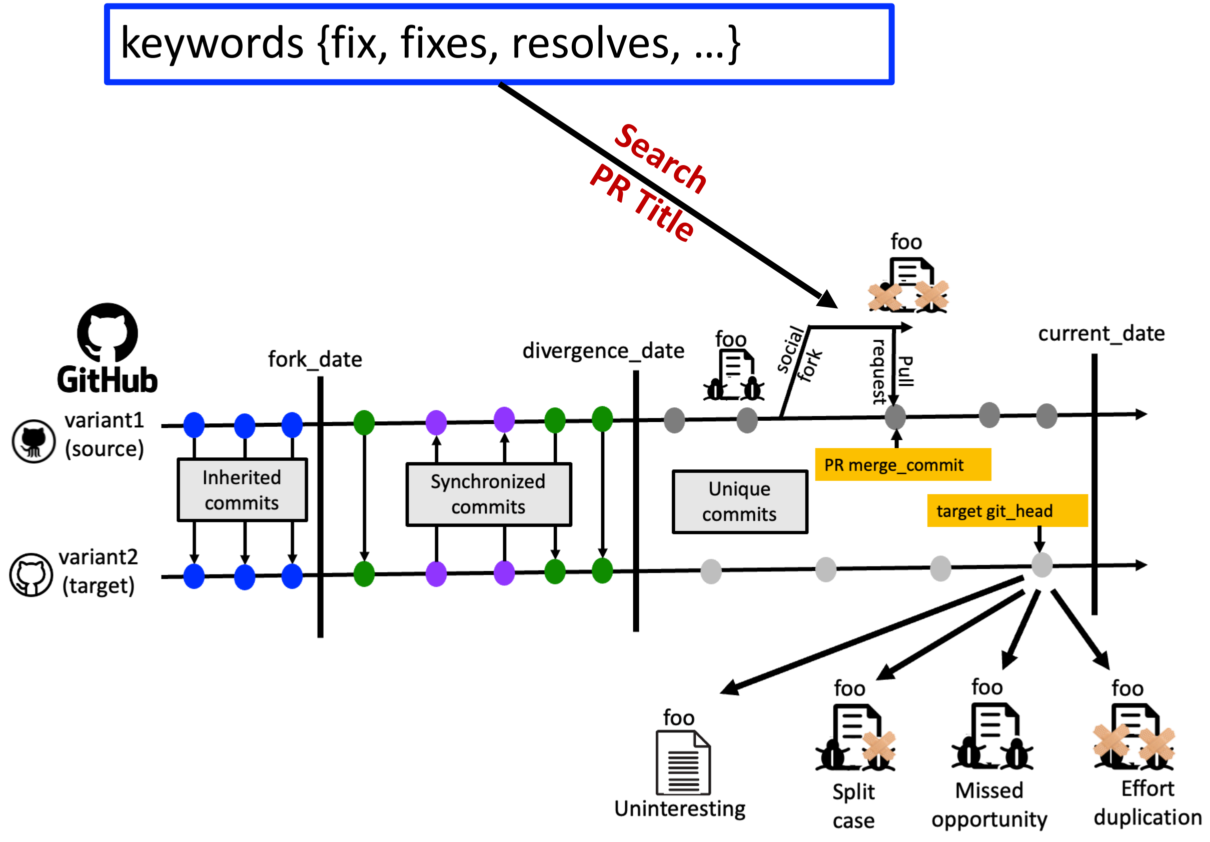
Buggy line

Research Questions

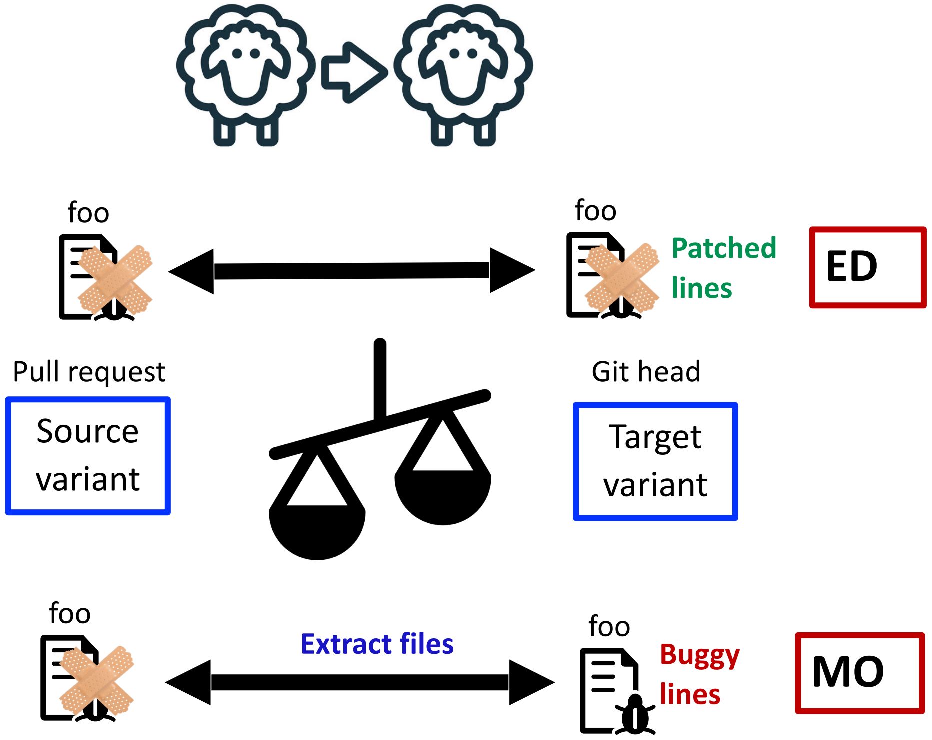
1. **RQ1:** How many cases of effort duplication and missed opportunities exist between divergent variants?

2. **RQ2:** How much patch technical lag exists between the source and target variants in divergent variants?

Method

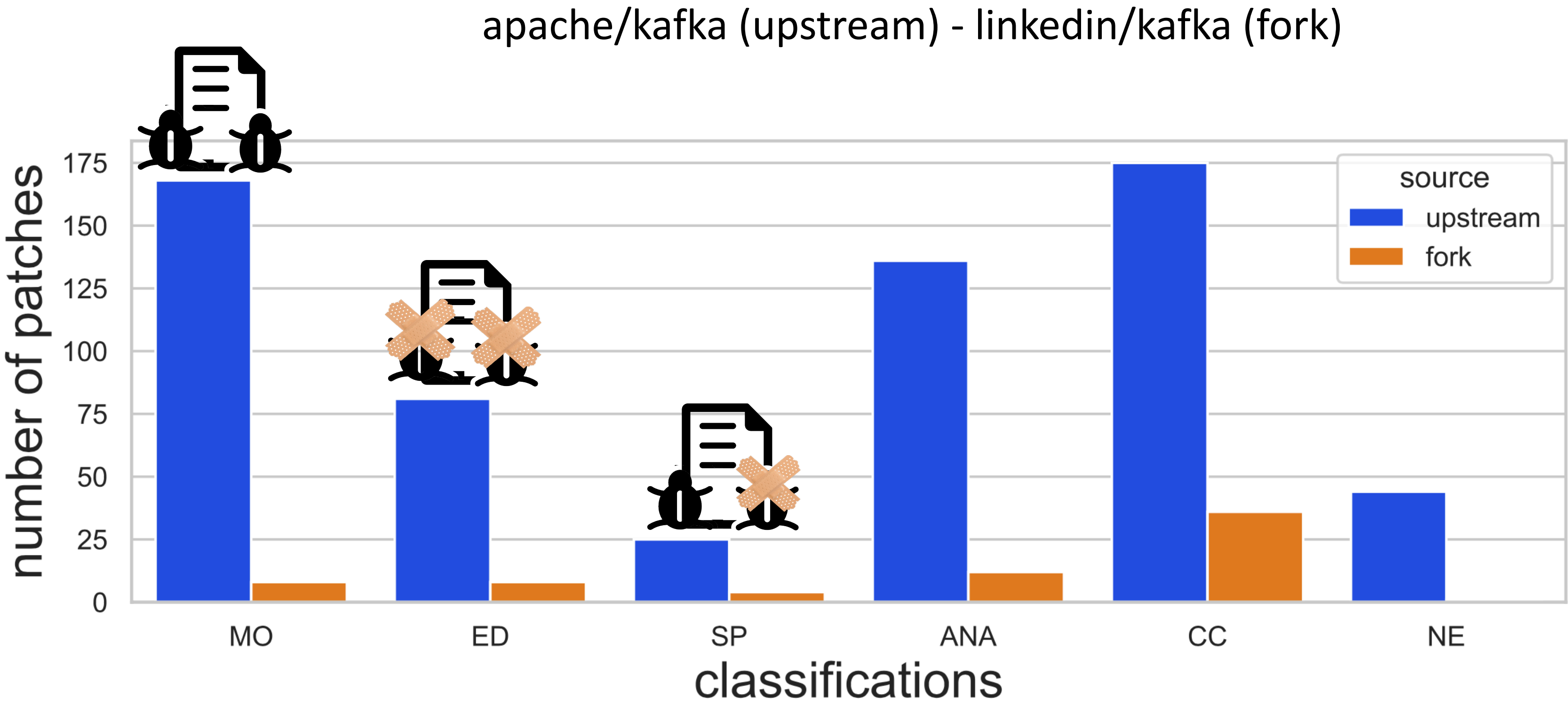


clone detection tool (PaReco)



Results

RQ1: How many cases of effort duplication and missed opportunities exist between divergent variants?



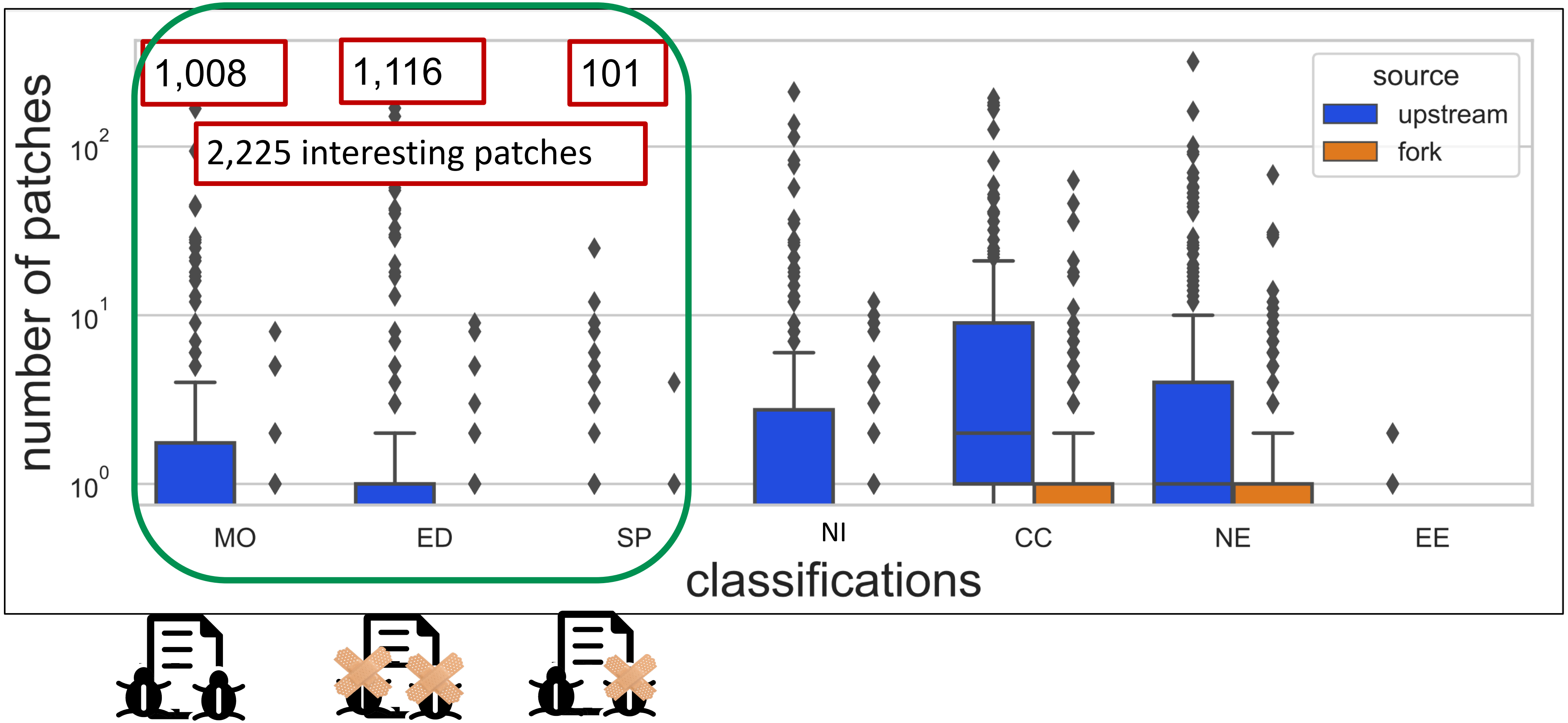
MO – Missed opportunity
ED – Effort duplication
SP – Both buggy and patched lines
NI – Uninteresting
CC – Unhandled programming language
NE – Missing file in target
EE – Error

Results

RQ1: How many cases of effort duplication and missed opportunities exist between divergent variants?

Precision	Recall	Accuracy	F1-Score
91.0%	80.2%	88.0%	85.3%

8,323 patches from 364 source variants



1,008

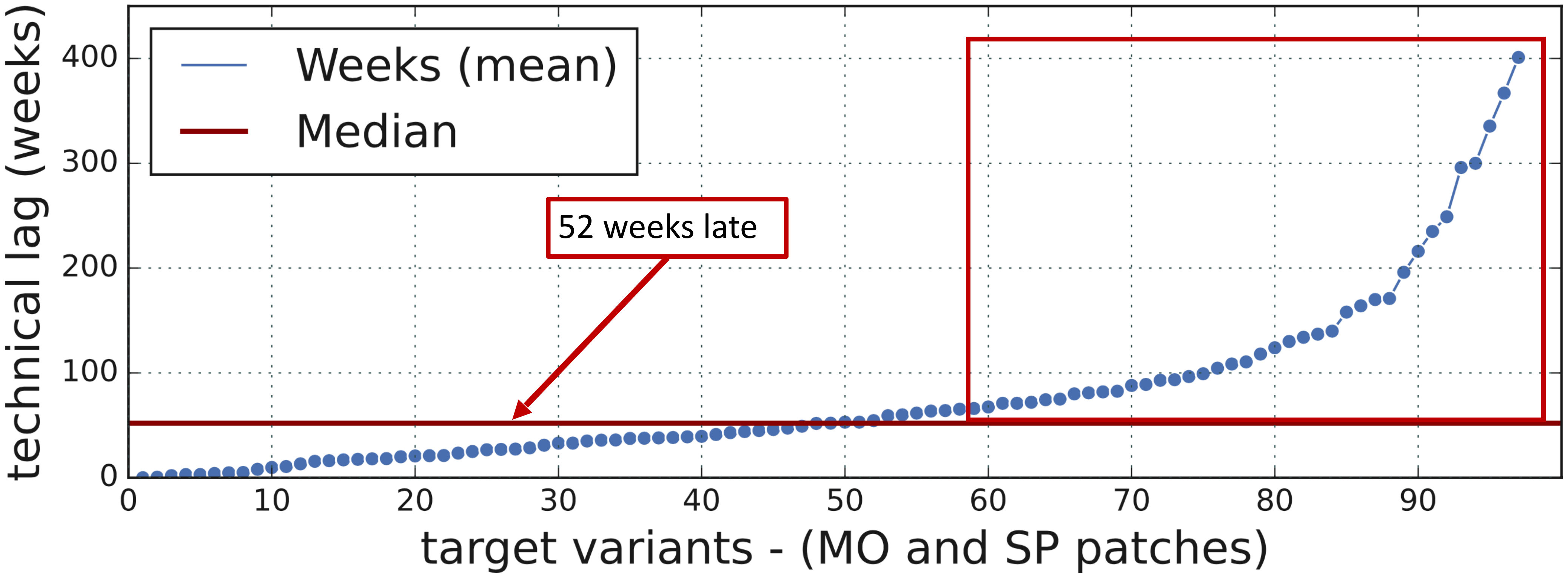
1,116

101

2,225 interesting patches

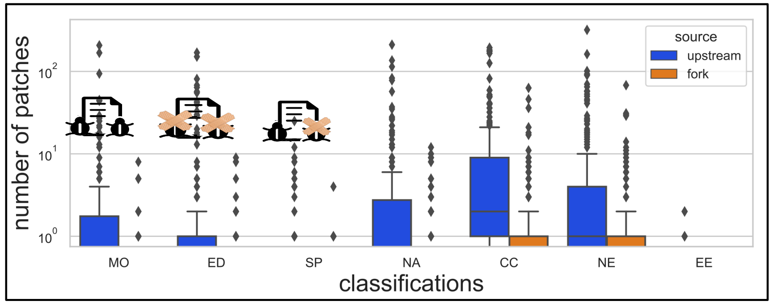
Results

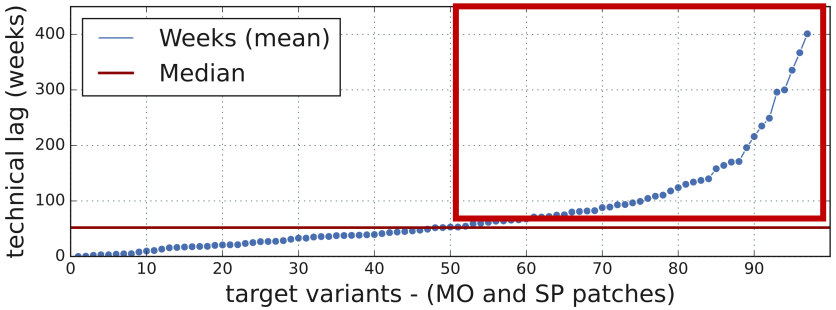
RQ2: How much patch technical lag exists between the source and target variants in divergent variants?

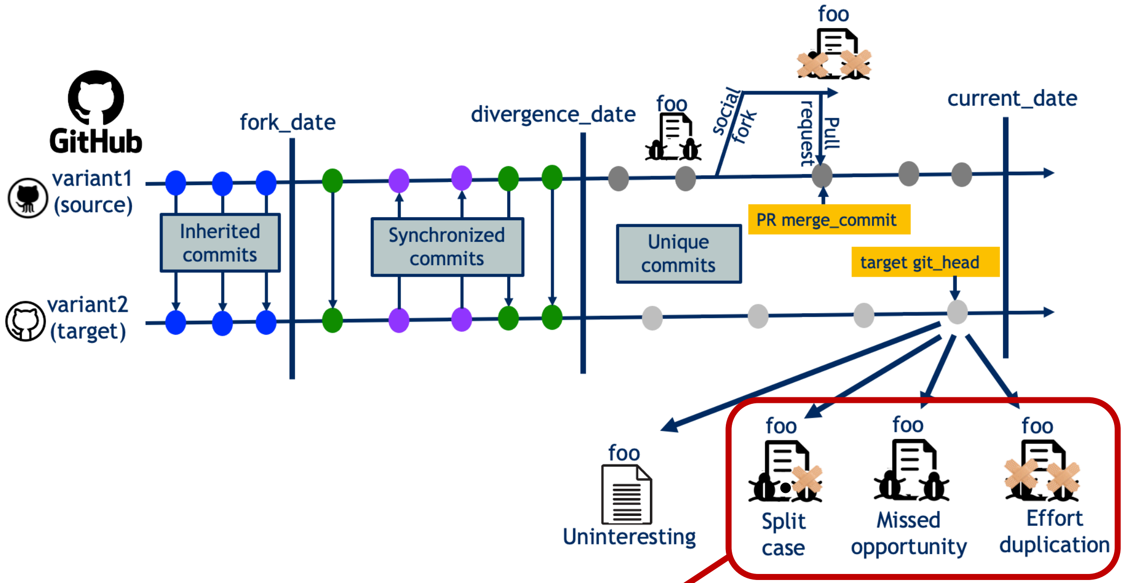


52 weeks late

What do we learn from the results?



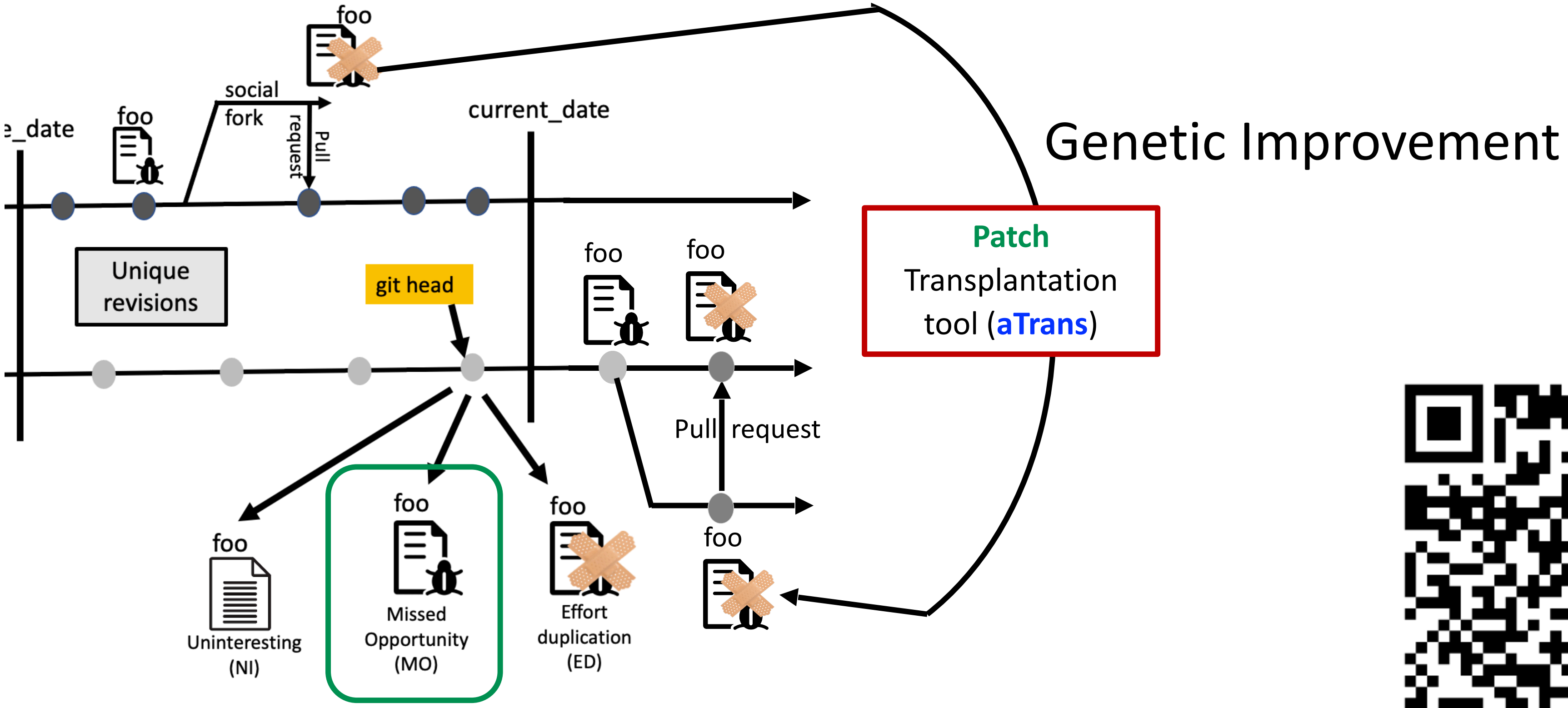




Variants on social coding platforms exhibit suboptimal maintenance

PaReco: Proof-of-Concept patch recommender tool

Current Work on PaReco



foo

social

fork

Pull request

e_date

foo

current_date

Genetic Improvement

Unique revisions

git head

foo

foo

Patch Transplantation tool (**aTrans**)

Pull request

foo

foo

foo

foo

Uninteresting (NI)

Missed Opportunity (MO)

Effort duplication (ED)

PaReco: Proof-of-Concept patch recommender tool