# Issues Characterization

omitted

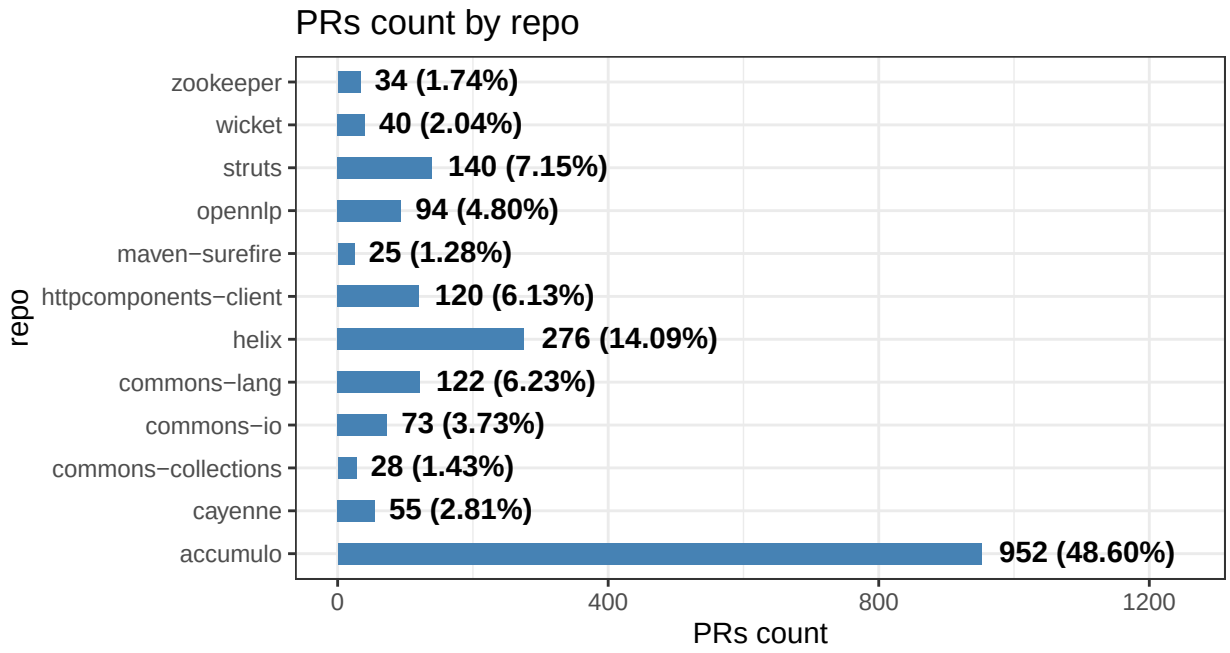
2024-04-05

## 1 Data overview

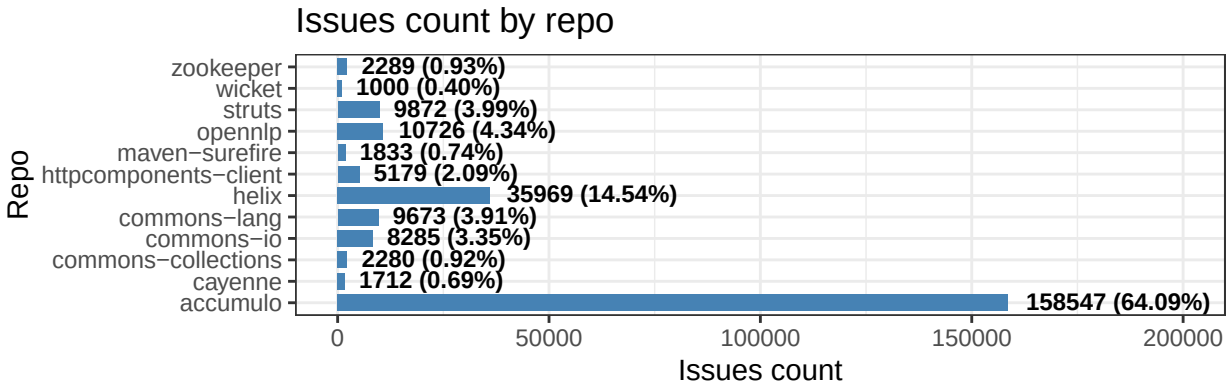
```
glimpse(all_data)
```

```
## Rows: 247,365
## Columns: 11
## $ repo                <chr> "accumulo", "accumulo", "accumulo", "accumulo", "a~
## $ pr_number           <int> 1302, 1302, 1302, 1302, 1302, 1302, 1302, 1302, 13~
## $ rule                <chr> "java:S1192", "java:S3776", "java:S1319", "java:S1~
## $ severity            <chr> "CRITICAL", "CRITICAL", "MINOR", "CRITICAL", "CRIT~
## $ file                <chr> "core/src/main/java/org/apache/accumulo/core/metad~
## $ type                <chr> "CODE_SMELL", "CODE_SMELL", "CODE_SMELL", "CODE_SM~
## $ status              <chr> "OPEN", "OPEN", "OPEN", "OPEN", "CLOSED", "OPEN", ~
## $ debt                <int> 8, 6, 10, 5, 6, 2, 5, 15, 30, 109, 30, 30, 5, 5, 5~
## $ ncloc_affected_file <int> 347, 347, 347, 347, 347, 347, 347, 347, 252, 252, ~
## $ complexity          <int> 68, 68, 68, 68, 68, 68, 68, 68, 86, 86, 86, 86, 86~
## $ origin              <chr> "NEW", "NEW", "NEW", "NEW", "PRE-EXISTING", "PRE-E~
```

### 1.1 PRs count by repo



### 1.2 Issues count by repo

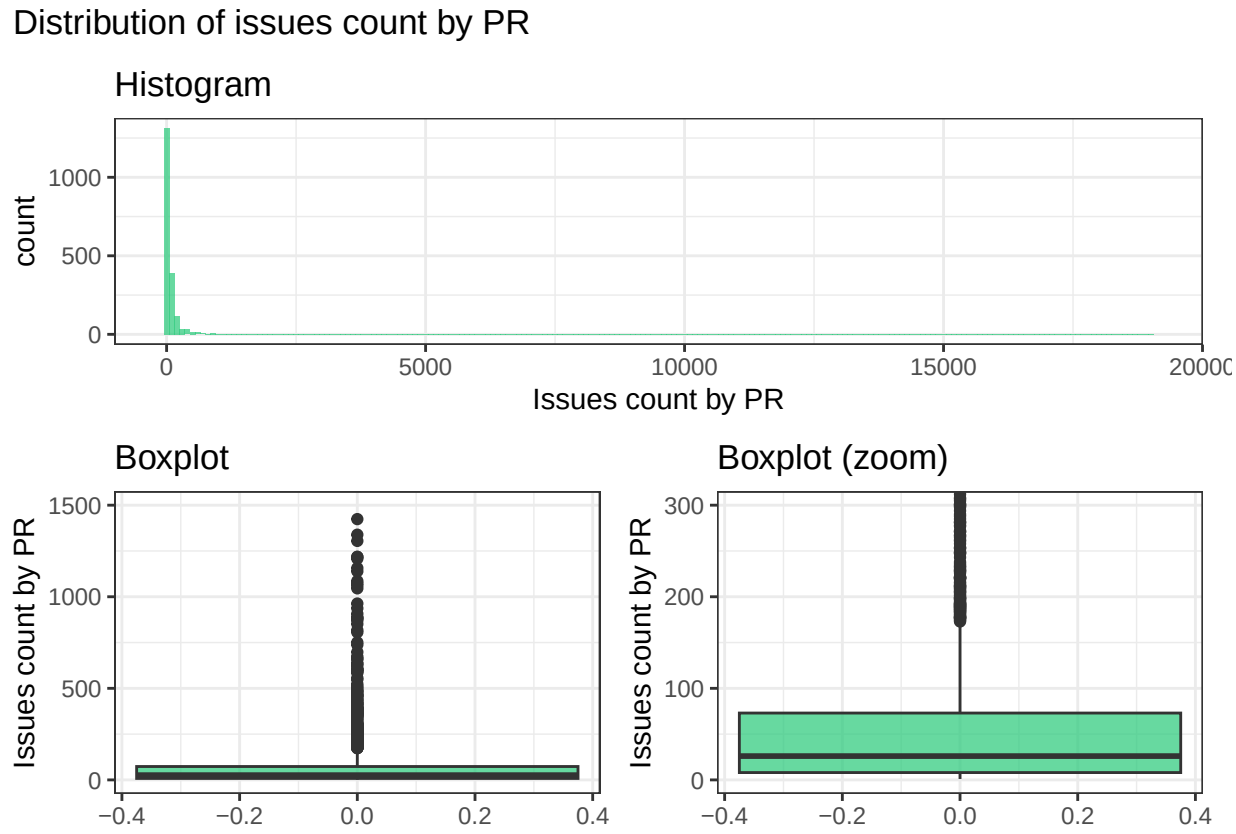



## 2 Issues

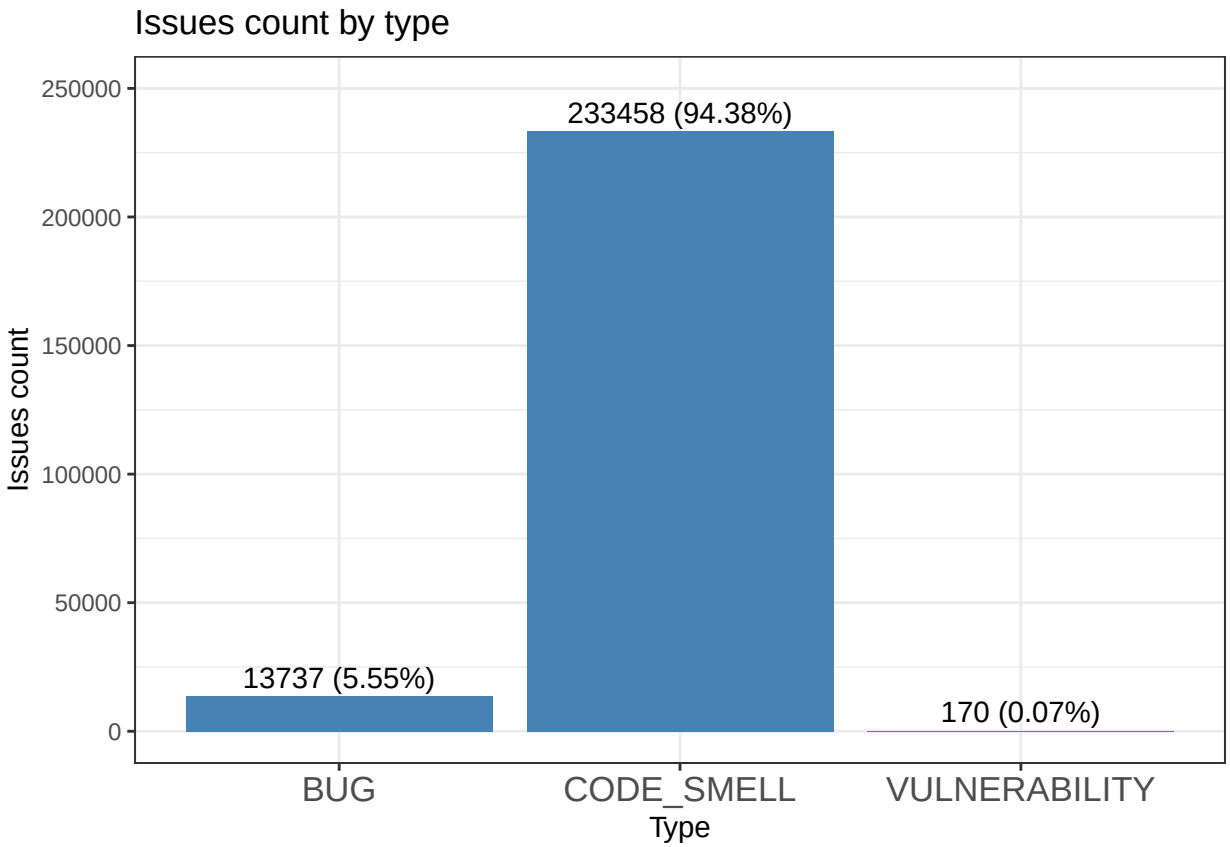
### 2.1 Issues count by PR

```
##     n_issues
##  Min.   :    1.0
##  1st Qu.:    8.0
##  Median :   26.0
##  Mean   :  126.3
##  3rd Qu.:   73.0
##  Max.   :18993.0
```

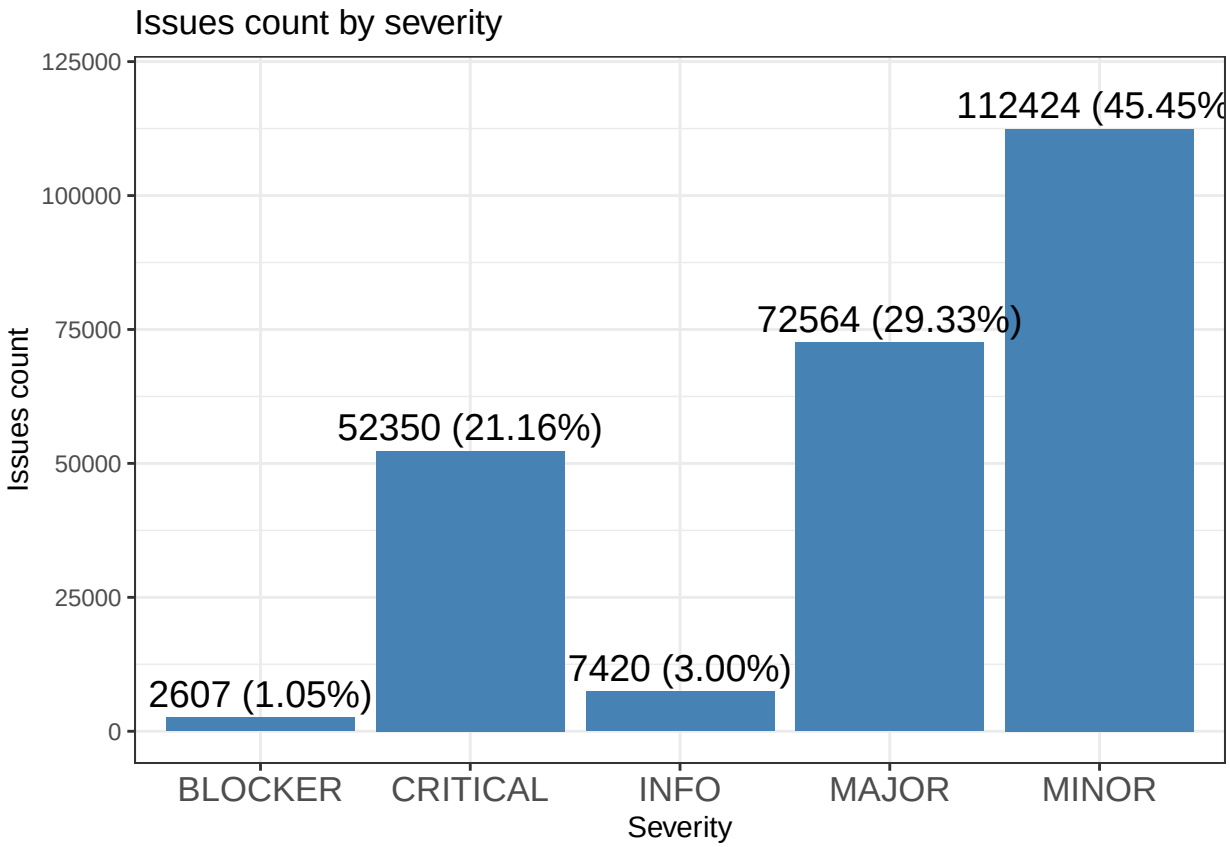
### 2.2 Distribution of issues count by PR



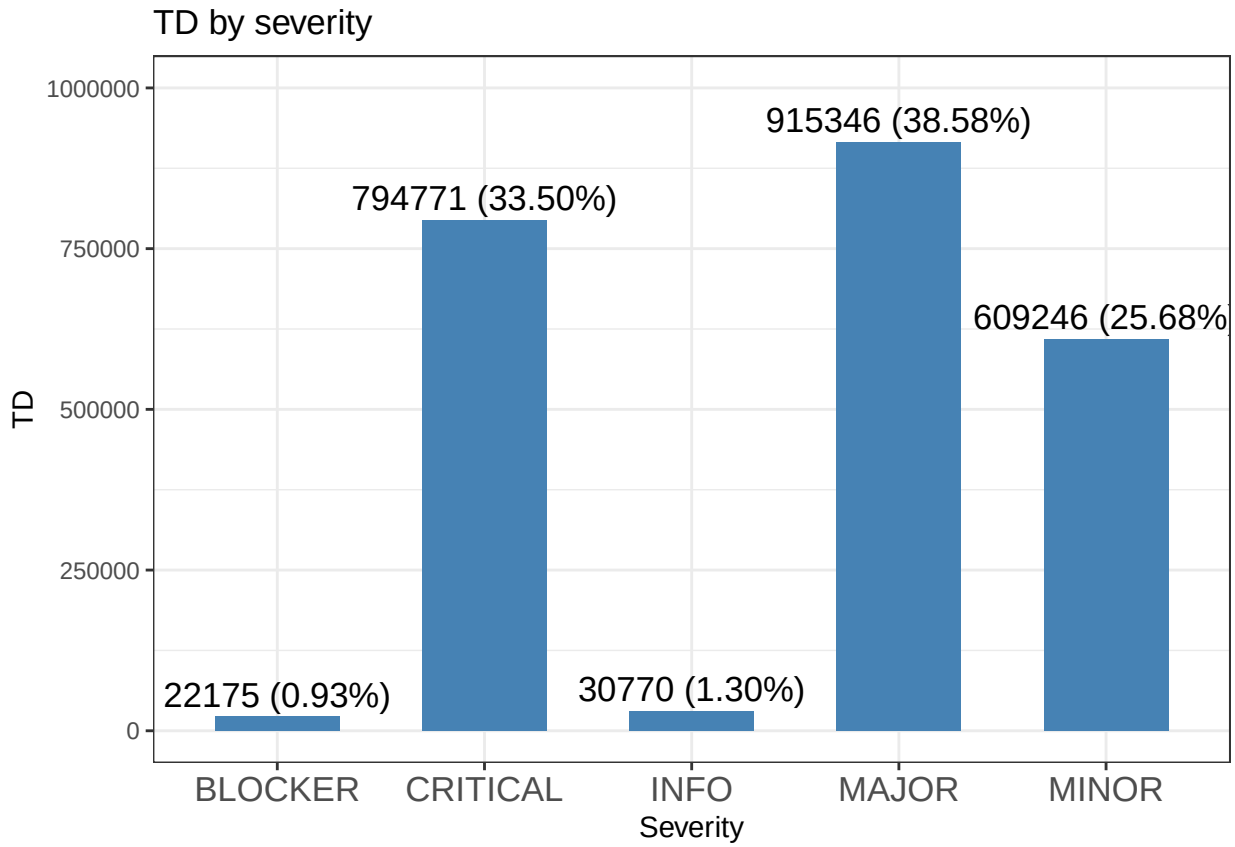
### 2.3 Issues count by type



### 2.4 Issues count by severity



### 2.5 TD by severity




### 2.6 Distinct rules count

```
##     n
## 1 264
```

264 different rules out of 622 (42.44%) were violated across the entire database.

### 2.7 Distinct resolved rules count

```
##     n
## 1 200
```

From 264 distinct rules detected, 202 types (76.51%) have at least one instance resolved.

### 2.8 Most frequent issues

```
## # A tibble: 10 x 3
##    rule           n percentage
##    <chr>      <int>      <dbl>
##  1 java:S1192 26850       10.9
##  2 java:S117  20425        8.26
```

Table 1: Top 10 most frequent issues

| Rule | Description | Type | Severity | Debt |
|---|---|---|---|---|
| S1192 | String literals should not be duplicated | CODE SMELL | Critical | 2min + 2min per additional instance |
| S117 | Local variable and method parameter names should comply with a naming convention | CODE SMELL | Minor | 2min |
| S2589 | Boolean expressions should not be gratuitous | CODE SMELL | Major | 10min |
| S116 | Field names should comply with a naming convention | CODE SMELL | Minor | 2min |
| S101 | Class names should comply with a naming convention | CODE SMELL | Minor | 5min |
| S112 | Generic exceptions should never be thrown | CODE SMELL | Major | 20min |
| S3776 | Cognitive Complexity of methods should not be too high | CODE SMELL | Critical | 5min + 1 min per point above threshold |
| S1125 | Boolean literals should not be redundant | CODE SMELL | Minor | 5min |
| S2293 | The diamond operator ("<>") should be used | CODE SMELL | Minor | 1min |
| S106 | Standard outputs should not be used directly to log anything | CODE SMELL | Major | 10min |

```
##  3 java:S2589 15764       6.37
##  4 java:S116  11624       4.70
##  5 java:S101  11506       4.65
##  6 java:S112  11444       4.63
##  7 java:S3776 11353       4.59
##  8 java:S1125 10537       4.26
##  9 java:S2293  9297       3.76
## 10 java:S106   6413       2.59
```

## 2.9 Most frequent issues by repo

```
## # A tibble: 120 x 3
## # Groups:   repo [12]
##    repo     rule           n
##    <chr>    <chr>      <int>
##  1 accumulo java:S117  17884
##  2 accumulo java:S2589 15657
##  3 accumulo java:S1192 13780
##  4 accumulo java:S101  11399
##  5 accumulo java:S112  10514
##  6 accumulo java:S1125 10223
##  7 accumulo java:S3776  8679
##  8 accumulo java:S2293  6591
##  9 accumulo java:S116   5097
## 10 accumulo java:S2142  5001
## # i 110 more rows
```

The TOP 10 most frequent issues represents 54.71% of total.

## 2.10 Less frequent issues

```
## # A tibble: 10 x 3
##    rule           n percentage
##    <chr>      <int>      <dbl>
##  1 java:S1217     1          0
```

Table 2: Top 10 less frequent issues

| Rule | Description | Type | Severity | Debt |
|---|---|---|---|---|
| S1217 | "Thread.run()" should not be called directly | BUG | Major | 20min |
| S1220 | The default unnamed package should not be used | CODE SMELL | Minor | 10min |
| S2110 | Invalid "Date" values should not be used | BUG | Major | 5min |
| S2121 | Silly String operations should not be made | BUG | Major | 5min |
| S2185 | Silly math should not be performed | CODE SMELL | Major | 15min |
| S2276 | "wait(...)" should be used instead of "Thread.sleep(...)" when a lock is held | BUG | Blocker | 5min |
| S2677 | "read" and "readLine" return values should be used | BUG | Major | 5min |
| S2885 | Non-thread-safe fields should not be static | BUG | Major | 15min |
| S3034 | Raw byte values should not be used in bitwise operations in combination with shifts | BUG | Major | 5min |
| S3923 | All branches in a conditional structure should not have exactly the same implementation | BUG | Major | 15min |

```
##  2 java:S1220     1           0
##  3 java:S2110     1           0
##  4 java:S2121     1           0
##  5 java:S2185     1           0
##  6 java:S2276     1           0
##  7 java:S2677     1           0
##  8 java:S2885     1           0
##  9 java:S3034     1           0
## 10 java:S3923     1           0
```

## 2.11 Issues by higher total TD

```
## # A tibble: 10 x 3
##    rule         debt percentage
##    <chr>       <int>      <dbl>
##  1 java:S1192 428812      18.1
##  2 java:S112  228880       9.65
##  3 java:S3776 214045       9.02
##  4 java:S2589 157640       6.64
##  5 java:S1874  92865       3.92
##  6 java:S2142  83865       3.54
##  7 java:S106   64130       2.70
##  8 java:S2157  59910       2.52
##  9 java:S101   57530       2.42
## 10 java:S1119  54420       2.29
```

The TOP 10 of issues by higher total TD represents 60.78% of total TD.

S1874 (major, obsolete), S2142 (major, error-handling), S2157 (critical, convention) and S1119 (major, confusing) are not in TOP 10 issues by count. S116 (minor, convention), S117 (minor, convention), S1125 (minor, clumsy) and S2293 (minor, clumsy) are not in TOP 10 by higher total TD.

The rules S101 (minor, convention), S106 (major, bad-practice), S112 (major, error-handling), S1192 (critical, design), S2589 (major, redundant) and S3776 (critical, brain-overload) are in both.

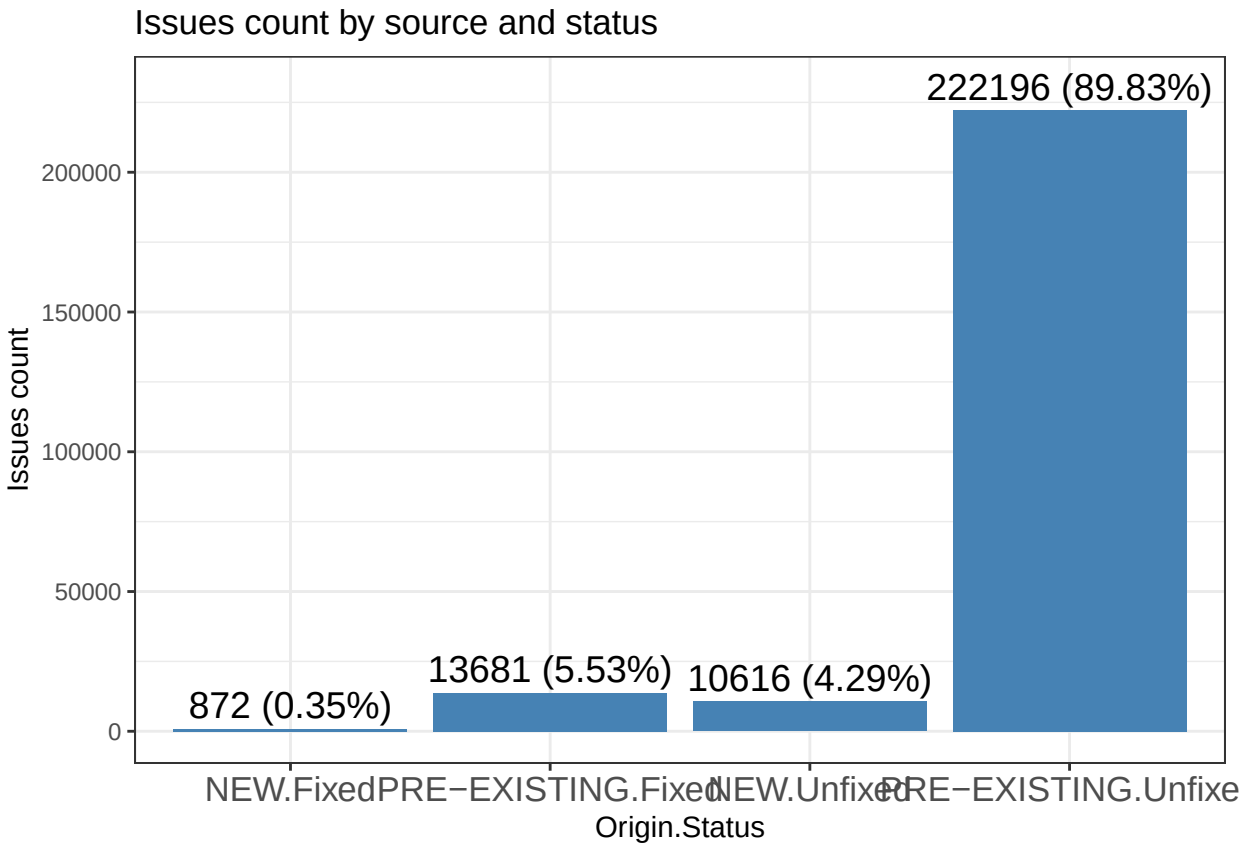
Table 3: Top 10 issues by total TD

| Rule | Description | Type | Severity | Debt |
|---|---|---|---|---|
| S1192 | String literals should not be duplicated | CODE SMELL | Critical | 2min + 2min per additional instance |
| S112 | Generic exceptions should never be thrown | CODE SMELL | Major | 20min |
| S3776 | Cognitive Complexity of methods should not be too high | CODE SMELL | Critical | 5min + 1 min per point above threshold |
| S2589 | Boolean expressions should not be gratuitous | CODE SMELL | Major | 10min |
| S1874 | "@Deprecated" code marked for removal should never be used | CODE SMELL | Major | 15min |
| S2142 | "InterruptedException" should not be ignored | BUG | Major | 15min |
| S106 | Standard outputs should not be used directly to log anything | CODE SMELL | Major | 10min |
| S2157 | "Cloneables" should implement "clone" | CODE SMELL | Critical | 30min |
| S101 | Class names should comply with a naming convention | CODE SMELL | Minor | 5min |
| S1119 | Labels should not be used | CODE SMELL | Major | 30min |

## 2.12 Issues by higher total TD and grouped by repo

```
## # A tibble: 120 x 4
## # Groups:   repo [12]
##    repo     rule         debt percentage
##    <chr>    <chr>       <int>      <dbl>
##  1 accumulo java:S1192 229912      15.4
##  2 accumulo java:S112  210280      14.1
##  3 accumulo java:S3776 159316      10.7
##  4 accumulo java:S2589 156570      10.5
##  5 accumulo java:S2142  75015       5.02
##  6 accumulo java:S2157  59700       4.00
##  7 accumulo java:S101   56995       3.82
##  8 accumulo java:S1125  51115       3.42
##  9 accumulo java:S1075  40120       2.68
## 10 accumulo java:S117   35768       2.39
## # i 110 more rows
```

### 2.13 Issues count by origin and status




## Distribution of issues count by origin and status

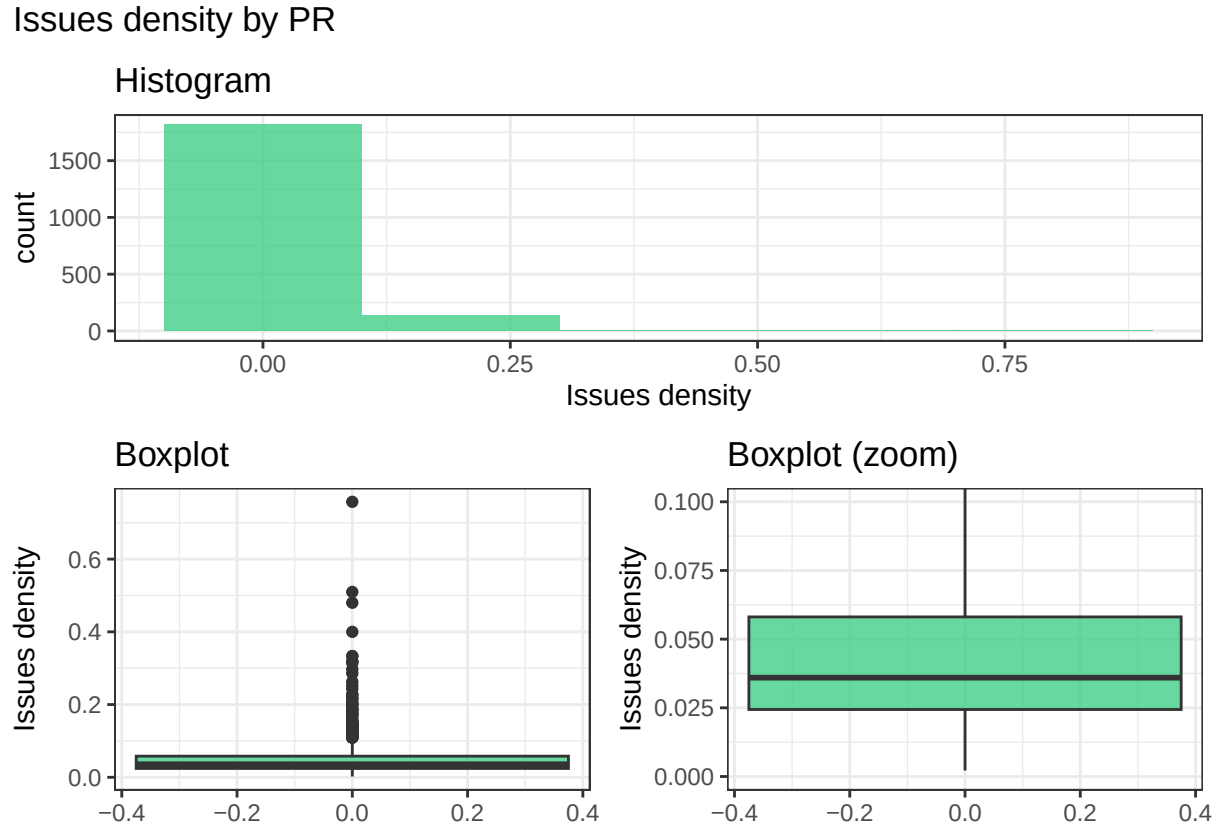
```
##      repo           n_issues_preexisting_unfixed n_issues_new_unfixed
##  Length:1959        Min.   :    0.0              Min.   :   0.000
##  Class :character   1st Qu.:    7.0              1st Qu.:   0.000
##  Mode  :character   Median :   21.0              Median :   0.000
##                     Mean   :  113.4              Mean   :   5.419
##                     3rd Qu.:   65.0              3rd Qu.:   2.000
##                     Max.   :18993.0              Max.   :2114.000
##  n_issues_preexisting_fixed n_issues_new_fixed
##  Min.   :   0.000           Min.   : 0.0000
##  1st Qu.:   0.000           1st Qu.: 0.0000
##  Median :   0.000           Median : 0.0000
##  Mean   :   6.984           Mean   : 0.4451
##  3rd Qu.:   2.000           3rd Qu.: 0.0000
##  Max.   :2910.000           Max.   :77.0000
```

### 2.14 Issues density (considering NCLOC from last commit)

Note: NCLOC is defined by SonarQube as the number of physical lines that contain at least one character that is not a white space, a tab, or part of a comment. Source.

```
##  issues_density
```

```
##  Min.   :0.002174
##  1st Qu.:0.024390
##  Median :0.035969
##  Mean   :0.047416
##  3rd Qu.:0.058095
##  Max.   :0.757576
```

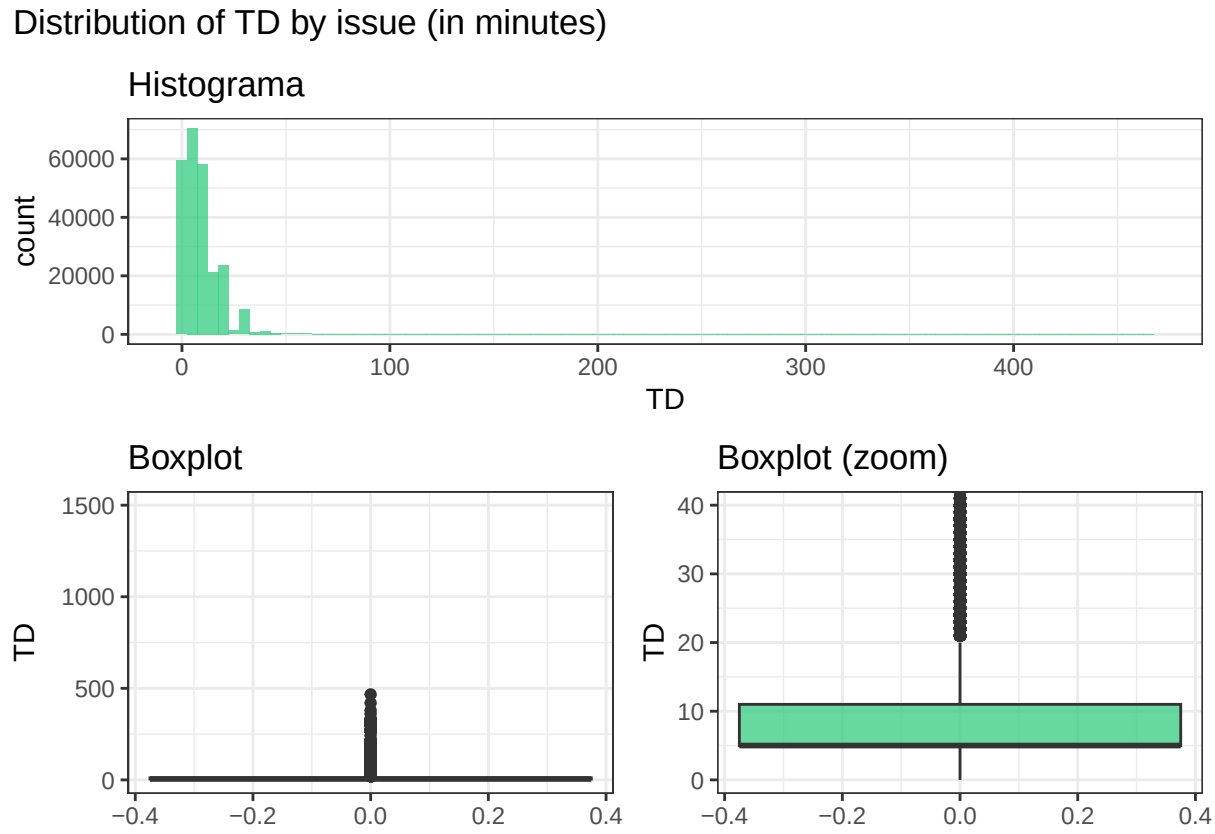



# 3 Technical Debt (TD)

## 3.1 Statistics of the TD by issue

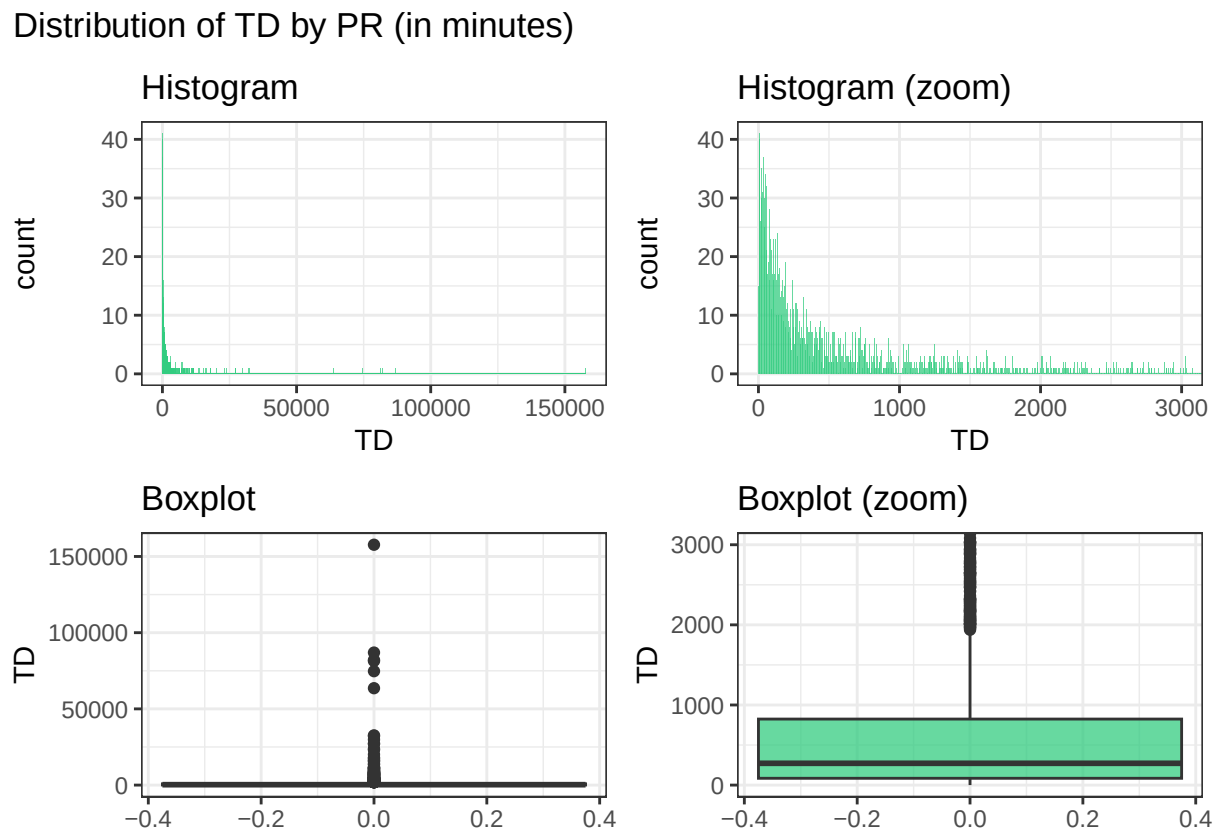
```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    0.00    5.00    5.00    9.59   11.00  467.00
```

### 3.2 Distribution of TD by issue



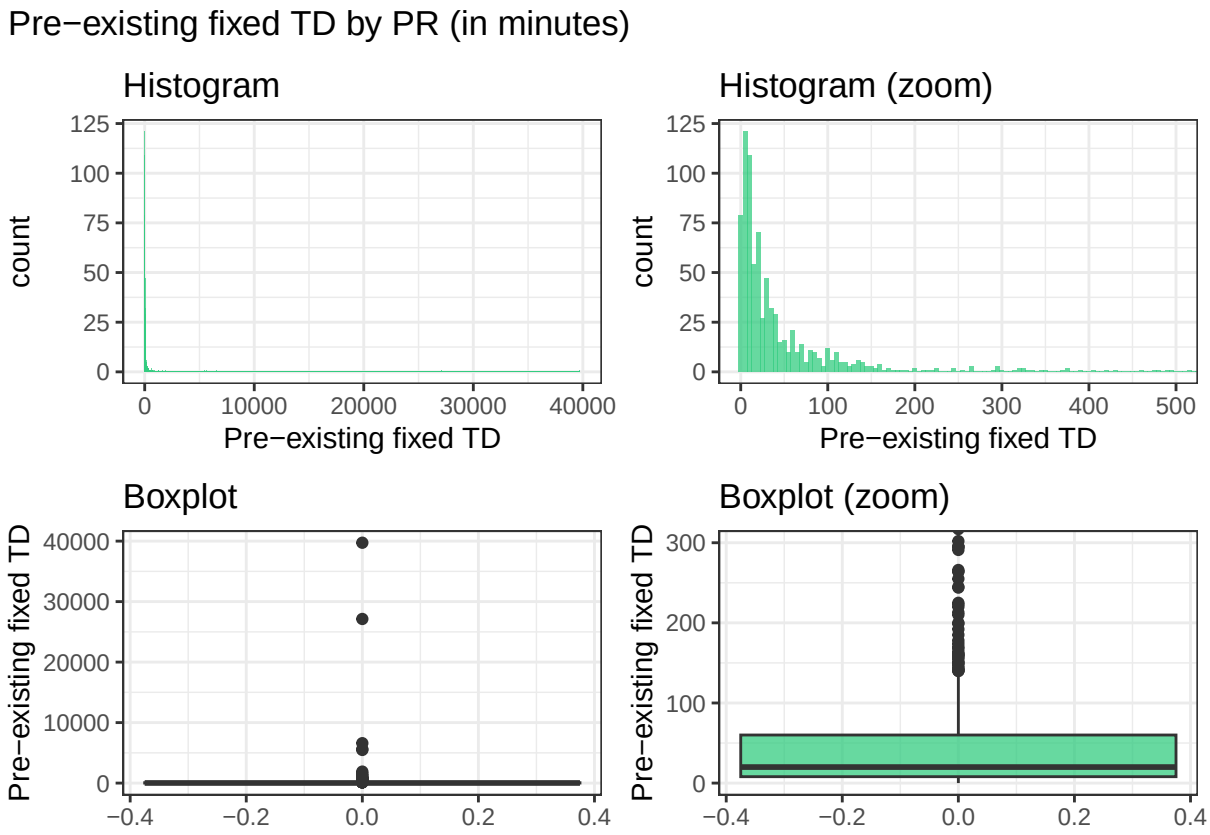

### 3.3 Distribution of TD by PR

```
##       debt
##  Min.   :     0
##  1st Qu.:    85
##  Median :   272
##  Mean   :  1211
##  3rd Qu.:   824
##  Max.   :157709
```



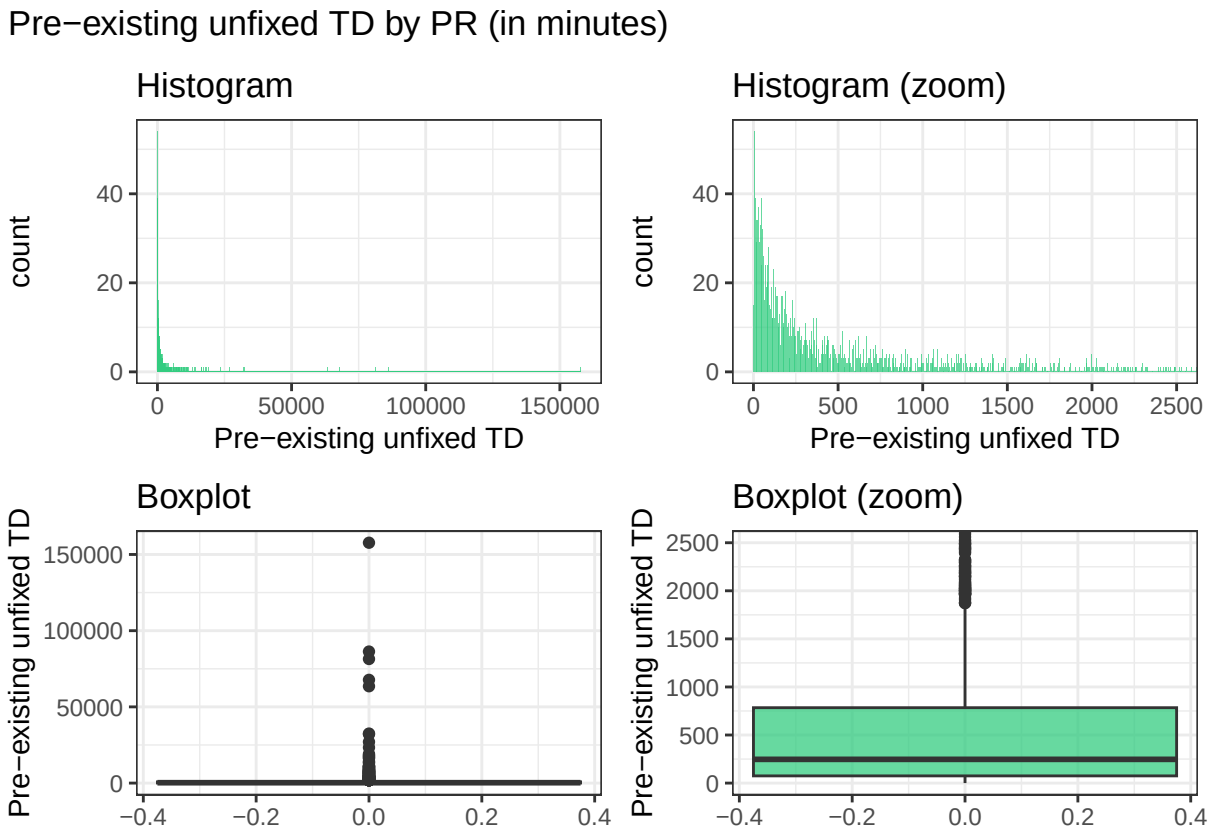

### 3.4 Distribution of pre-existing fixed TD by PR

```
##  preexisting_fixed_debt
##  Min.   :    0.0
##  1st Qu.:    8.0
##  Median :   20.0
##  Mean   :  166.2
##  3rd Qu.:   60.0
##  Max.   :39735.0
```




### 3.5 Distribution of pre-existing unfixed TD by PR

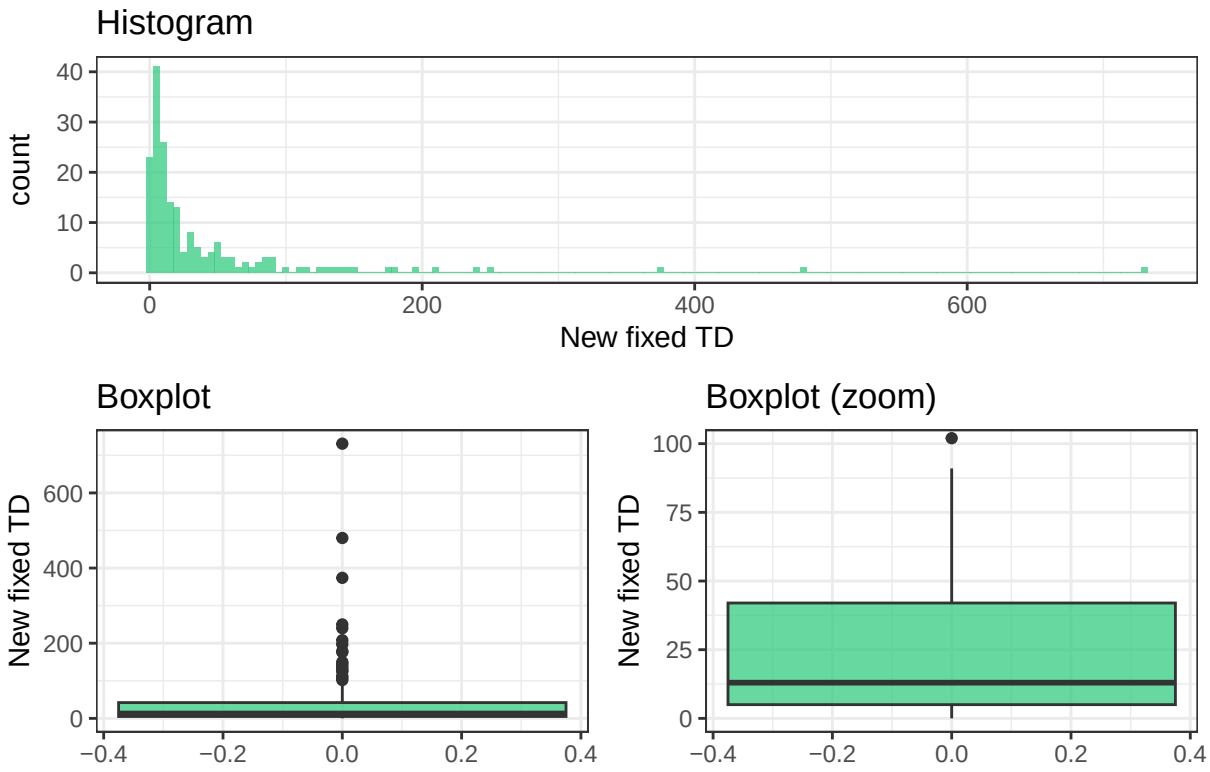
```
##  preexisting_unfixed_debt
##  Min.   :     0
##  1st Qu.:    75
##  Median :   247
##  Mean   :  1116
##  3rd Qu.:   784
##  Max.   :157709
```




### 3.6 Distribution of new fixed TD by PR

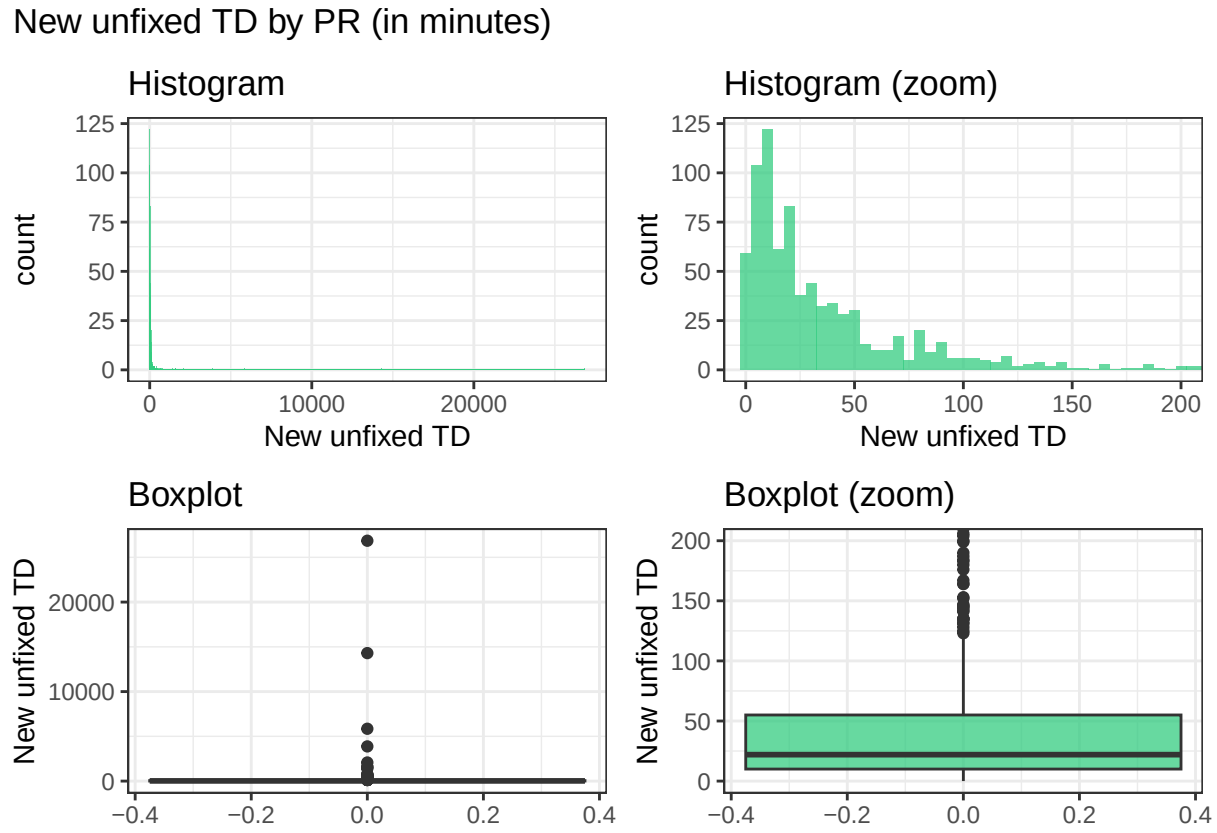
```
##  new_fixed_debt
##  Min.   :  0.00
##  1st Qu.:  5.00
##  Median : 13.00
##  Mean   : 40.17
##  3rd Qu.: 42.00
##  Max.   :731.00
```

New fixed TD by PR (in minutes)




### 3.7 Distribution of new unfixed TD by PR

```
##  new_unfixed_debt
##  Min.   :    0.0
##  1st Qu.:   10.0
##  Median :   22.0
##  Mean   :  119.9
##  3rd Qu.:   55.0
##  Max.   :26855.0
```




# 4 Which projects use ASATs?

| Project | ASATs |
|---|---|
| accumulo | CheckStyle, FindBugs and SpotBugs |
| cayenne | - |
| commons-collections | CheckStyle, SpotBugs and PMD |
| commons-io | CheckStyle, FindBugs and PMD |
| commons-lang | CheckStyle, FindBugs, Spotbugs and PMD |
| helix | CheckStyle |
| httpcomponents-client | CheckStyle |
| maven-surefire | CheckStyle and FindBugs |
| opennlp | CheckStyle |
| struts | SonarQube |
| wicket | FindBugs |
| zookeeper | CheckStyle, FindBugs and SpotBugs |

# 5 Most common coding rules by issue types

## 5.1 Pre-existing unfixed issues

```
## # A tibble: 10 x 4
```

```
##    rule       severity     n percentage
##    <chr>      <chr>    <int>      <dbl>
##  1 java:S1192 CRITICAL 24562      11.1
##  2 java:S117  MINOR    17853       8.04
##  3 java:S2589 MAJOR    14783       6.65
##  4 java:S116  MINOR    10853       4.88
##  5 java:S101  MINOR    10718       4.82
##  6 java:S112  MAJOR    10563       4.75
##  7 java:S3776 CRITICAL 10434       4.70
##  8 java:S1125 MINOR     9774       4.40
##  9 java:S2293 MINOR     8430       3.79
## 10 java:S106  MAJOR     6040       2.72
```

### 5.2 Pre-existing fixed issues

```
## # A tibble: 10 x 4
##    rule       severity     n percentage
##    <chr>      <chr>    <int>      <dbl>
##  1 java:S117  MINOR     1162       8.49
##  2 java:S1192 CRITICAL  1030       7.53
##  3 java:S1119 MAJOR      843       6.16
##  4 java:S3740 MAJOR      698       5.10
##  5 java:S112  MAJOR      569       4.16
##  6 java:S3776 CRITICAL   520       3.80
##  7 java:S100  MINOR      455       3.33
##  8 java:S1874 MINOR      443       3.24
##  9 java:S1161 MAJOR      422       3.08
## 10 java:S2293 MINOR      418       3.06
```

### 5.3 New unfixed issues

```
## # A tibble: 10 x 4
##    rule       severity     n percentage
##    <chr>      <chr>    <int>      <dbl>
##  1 java:S117  MINOR     1398      13.2
##  2 java:S1192 CRITICAL  1202      11.3
##  3 java:S1119 MAJOR      644       6.07
##  4 java:S2589 MAJOR      602       5.67
##  5 java:S101  MINOR      518       4.88
##  6 java:S116  MINOR      480       4.52
##  7 java:S1125 MINOR      468       4.41
##  8 java:S2293 MINOR      418       3.94
##  9 java:S3776 CRITICAL   368       3.47
## 10 java:S112  MAJOR      274       2.58
```

### 5.4 New fixed issues

```
## # A tibble: 10 x 4
##    rule       severity     n percentage
##    <chr>      <chr>    <int>      <dbl>
##  1 java:S1135 INFO        67       7.68
##  2 java:S1192 CRITICAL    56       6.42
```

Table 5: Issues description.

| Rule | Description | Type | Severity | Debt | Issue Class. |
|---|---|---|---|---|---|
| S100 | Method names should comply with a naming convention | CODE SMELL | Minor | 5min | PF |
| S101 | Class names should comply with a naming convention | CODE SMELL | Minor | 5min | PF + NU |
| S106 | Standard outputs should not be used directly to log anything | CODE SMELL | Major | 10 min | PU + NF |
| S112 | Generic exceptions should never be thrown | CODE SMELL | Major | 20min | PU + PF + NU + NF |
| S116 | Field names should comply with a naming convention | CODE SMELL | Minor | 2min | PU + NU + NF |
| S117 | Local variable and method parameter names should comply with a naming convention | CODE SMELL | Minor | 2min | PU + PF + NU |
| S1119 | Labels should not be used | CODE SMELL | MAJOR | 30min | PF + NU |
| S1125 | Boolean literals should not be redundant | CODE SMELL | Minor | 5min | PU + NU |
| S1135 | Track uses of "TODO" tags | CODE SMELL | Info | 0min | NF |
| S1161 | "@Override" should be used on overriding and implementing methods | CODE SMELL | MAJOR | 5min | PF |
| S1192 | String literals should not be duplicated | CODE SMELL | Critical | 2min + 2min per duplicated instance | PU + PF + NU + NF |
| S1874 | "@Deprecated" code should not be used | CODE SMELL | Minor | 15min | PF + NF |
| S2259 | Null pointers should not be dereferenced | BUG | Major | 10min | NF |
| S2293 | The diamond operator ("<>") should be used | CODE SMELL | Minor | 1min | PU + PF + NU + NF |
| S2589 | Boolean expressions should not be gratuitous | CODE SMELL | Major | 10min | PU + NU + NF |
| S3740 | Raw types should not be used | CODE SMELL | Major | 5min | PF |
| S3776 | Cognitive Complexity of methods should not be too high | CODE SMELL | Critical | 5min + 1min per point over the threshold | PU + PF + NU + NF |

```
##  3 java:S1874 MINOR       55       6.31
##  4 java:S106  MAJOR       38       4.36
##  5 java:S112  MAJOR       38       4.36
##  6 java:S2293 MINOR       31       3.56
##  7 java:S3776 CRITICAL    31       3.56
##  8 java:S116  MINOR       30       3.44
##  9 java:S2259 MAJOR       29       3.33
## 10 java:S2589 MAJOR       28       3.21
```

## 5.5 Table with the issues description

Legend: PF: Pre-existing fixed; PU: Pre-existing unfixed; NF: New fixed; NU: New unfixed.

## 5.6 Percentage of severity of the new unfixed issues

```
## # A tibble: 5 x 3
```

```
##   severity     n percentage
##   <chr>    <int>      <dbl>
## 1 MINOR     5179      48.8
## 2 MAJOR     2747      25.9
## 3 CRITICAL  2135      20.1
## 4 INFO       438       4.13
## 5 BLOCKER    117       1.10
```

# 6 Outliers

## 6.1 Issues count by PR

```
## # A tibble: 3 x 3
##   repo     pr_number n_issues
##   <chr>        <int>    <int>
## 1 accumulo      1433    18993
## 2 accumulo      2966    12878
## 3 accumulo      2706    12081
```

The PR 1433 of accumulo modifies 1,995 files within the PR, 2,132 files in the branch, and 55,972 lines in the branch (30,103 additions + 25,869 deletions). This large number of modified files is explained because PR modifies the license header standard, which exists in all files.

## 6.2 Issues density

### 6.2.1 Lower

```
## # A tibble: 10 x 5
##    repo                  pr_number ncloc smell_density n_issues
##    <chr>                     <int> <int>         <dbl>    <int>
##  1 httpcomponents-client       255   460       0.00217        1
##  2 accumulo                   2932   459       0.00218        1
##  3 accumulo                   1566   403       0.00248        1
##  4 opennlp                     550   386       0.00259        1
##  5 httpcomponents-client       221   275       0.00364        1
##  6 httpcomponents-client       118   246       0.00407        1
##  7 commons-lang                380   486       0.00412        2
##  8 commons-lang                556   486       0.00412        2
##  9 commons-lang               1041   482       0.00415        2
## 10 httpcomponents-client       225   232       0.00431        1
```

The PR 255 of httpcomponents-client has only one violation for 460 of NCLOC. It modifies only 1 file and 2 lines in the branch.

### 6.2.2 Higher

```
## # A tibble: 10 x 4
##    repo     pr_number ncloc smell_density
##    <chr>        <int> <int>         <dbl>
##  1 helix         1975    33         0.758
```

```
##  2 helix         2488    51         0.510
##  3 accumulo      4215    25         0.48
##  4 accumulo      3612   215         0.4
##  5 helix         2421     3         0.333
##  6 helix         2517     3         0.333
##  7 accumulo      2922    69         0.319
##  8 opennlp        561 17056         0.318
##  9 helix         2549   140         0.314
## 10 opennlp        563   232         0.297
```

The PR 1975 of helix modifies one file and 9 lines (9 additions). This one file has 33 of NCLOC and it has 25 issues in this file, so 25 issues in only 33 lines. The issues violate the coding rules S1104 (8 issues, class variable fields should not have public accessibility), S1118 (1 issue, utility classes should not have public constructors), S1444 (8 issues, "public static" fields should be constant) and S3008 (8 issues, Static non-final field names should comply with a naming convention). Most issues are of MINOR severity and are related to attribute conventions, each attribute of the class in question violates rules S1104, S1444 and S3008 simultaneously.

## 6.3 Issues TD

### 6.3.1 Lower

```
##   debt       rule
## 1    0 java:S1135
## 2    0 java:S1134
## 3    1 java:S3626
```

- S1134: Track uses of "FIXME" tags. Just a way that SonarQube uses to help track usage of FIXME tags, so it doesn't put an associated technical debt;
- S1135: Track uses of "TODO" tags. Same motivation as before.

### 6.3.2 Higher

```
##    debt       rule
## 1   467 java:S3776
## 2   420  java:S110
## 3   380  java:S135
## 4   360  java:S110
## 5   330  java:S110
## 6   330  java:S110
## 7   330  java:S110
## 8   330  java:S110
## 9   330  java:S110
## 10  330  java:S110
```

The rules S3776, S135 and S110 are the ones that present the most TD in a single instance (up to 467 minutes). The three rules have cumulative effort that change with each instance.