Lightweight Semantic Conflict Detection with Static Analysis
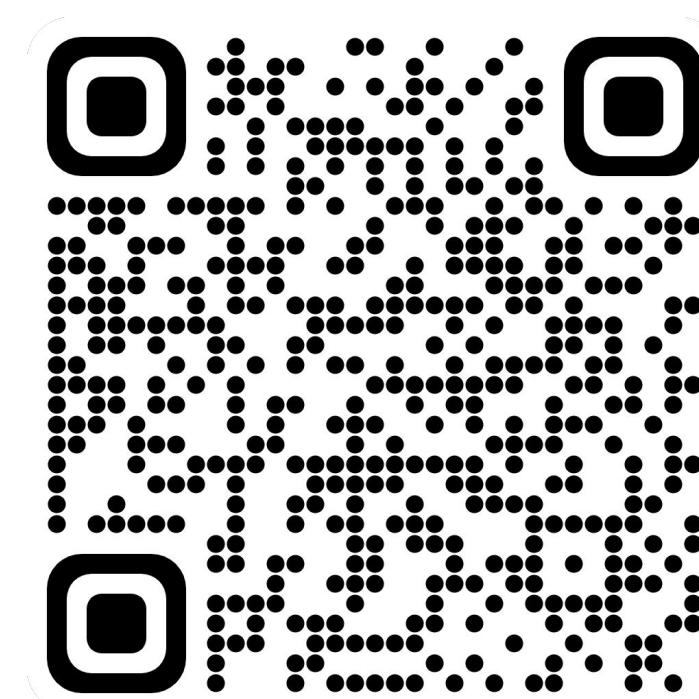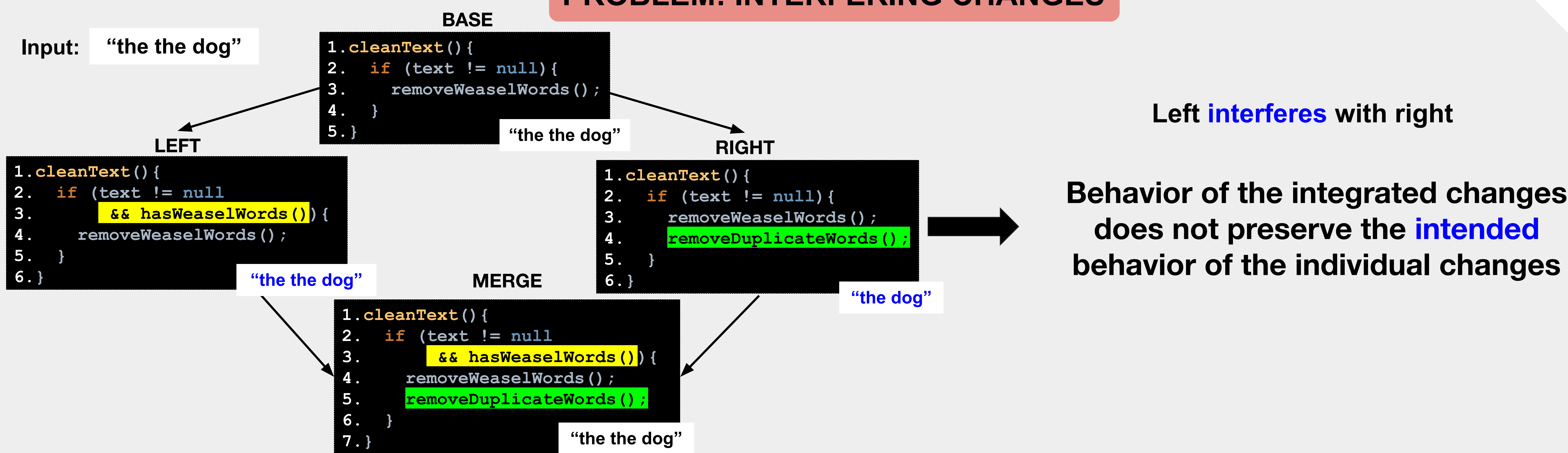
Galileu Santos de Jesus[1]   Paulo Borba[1]
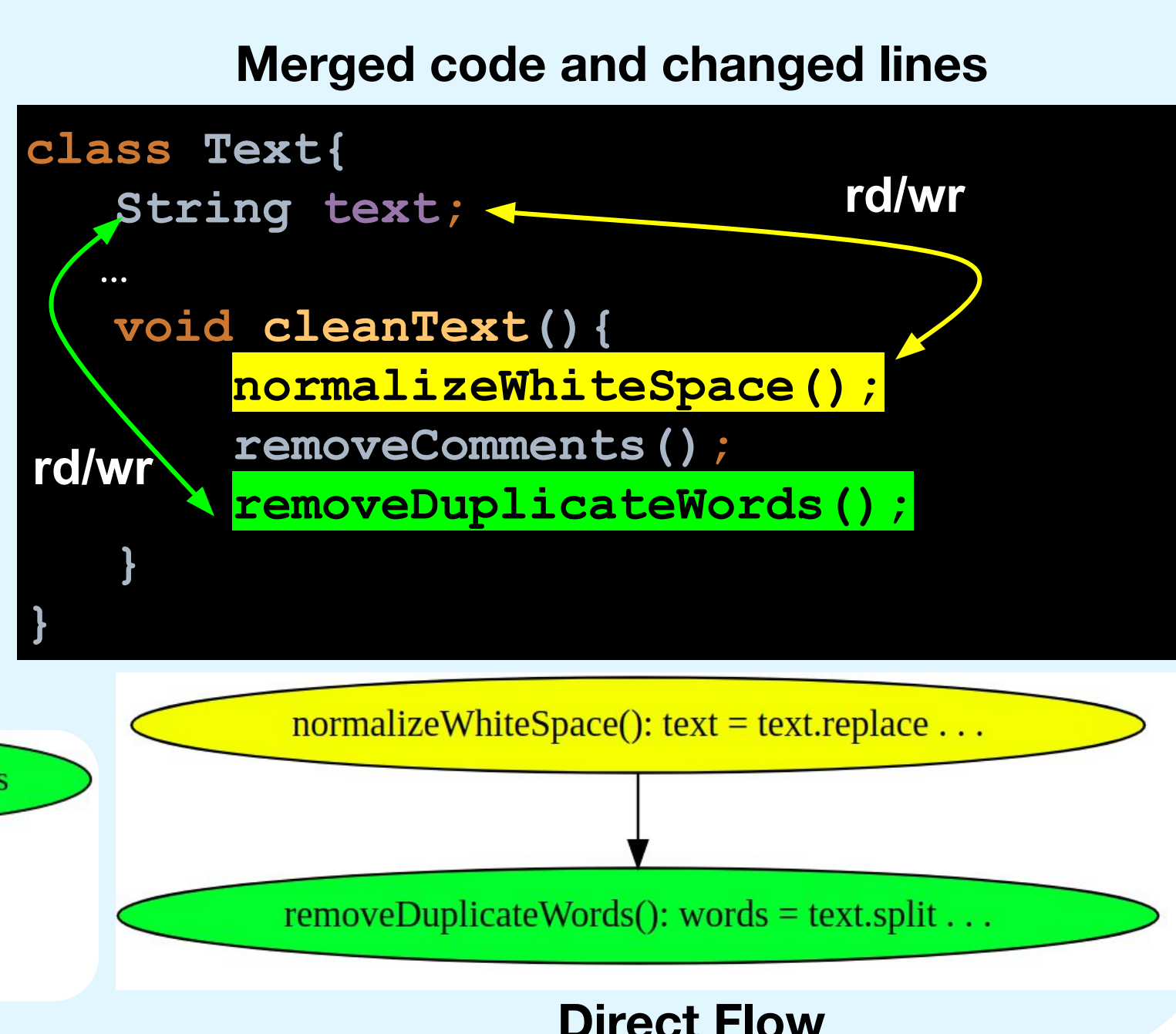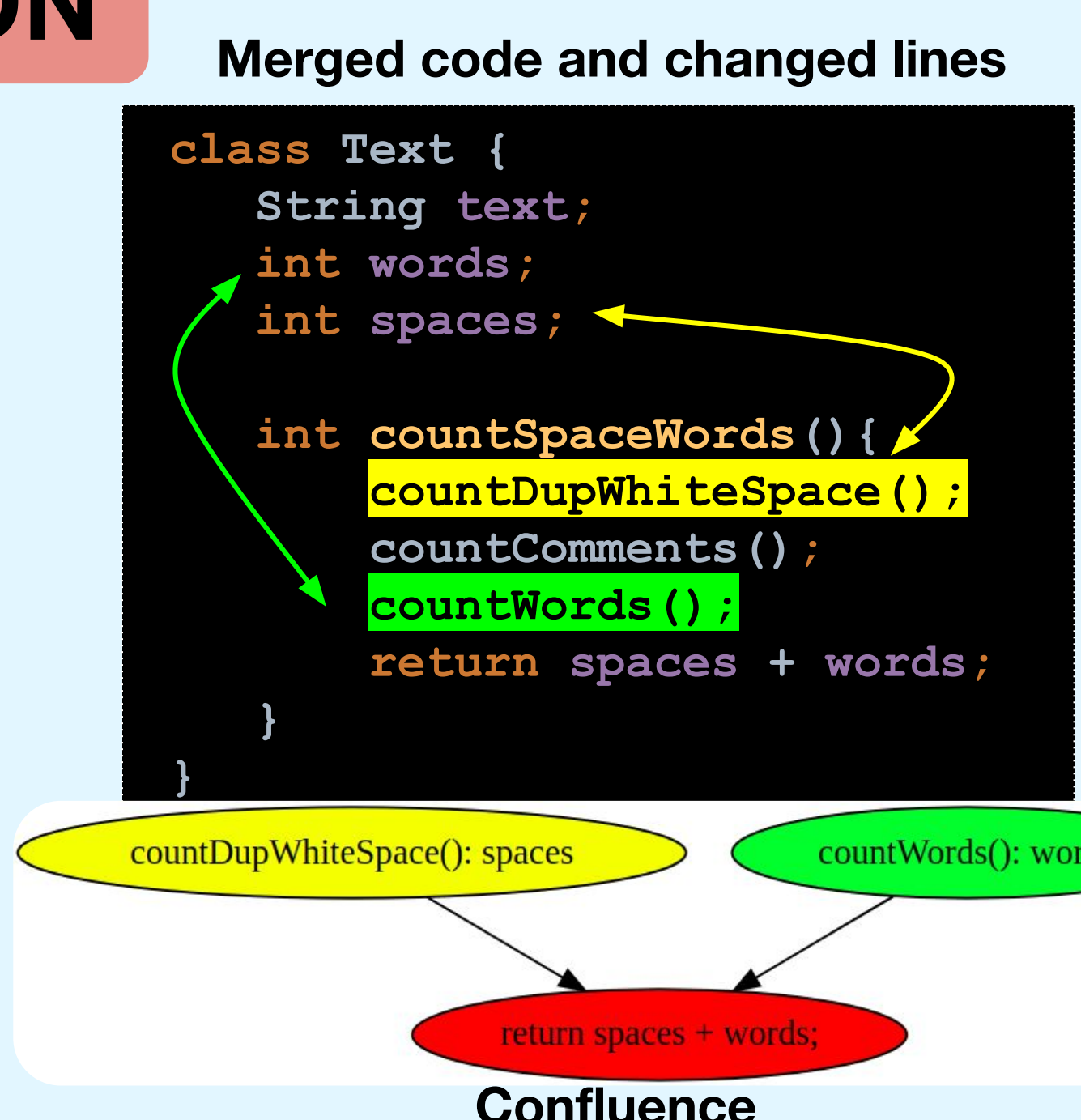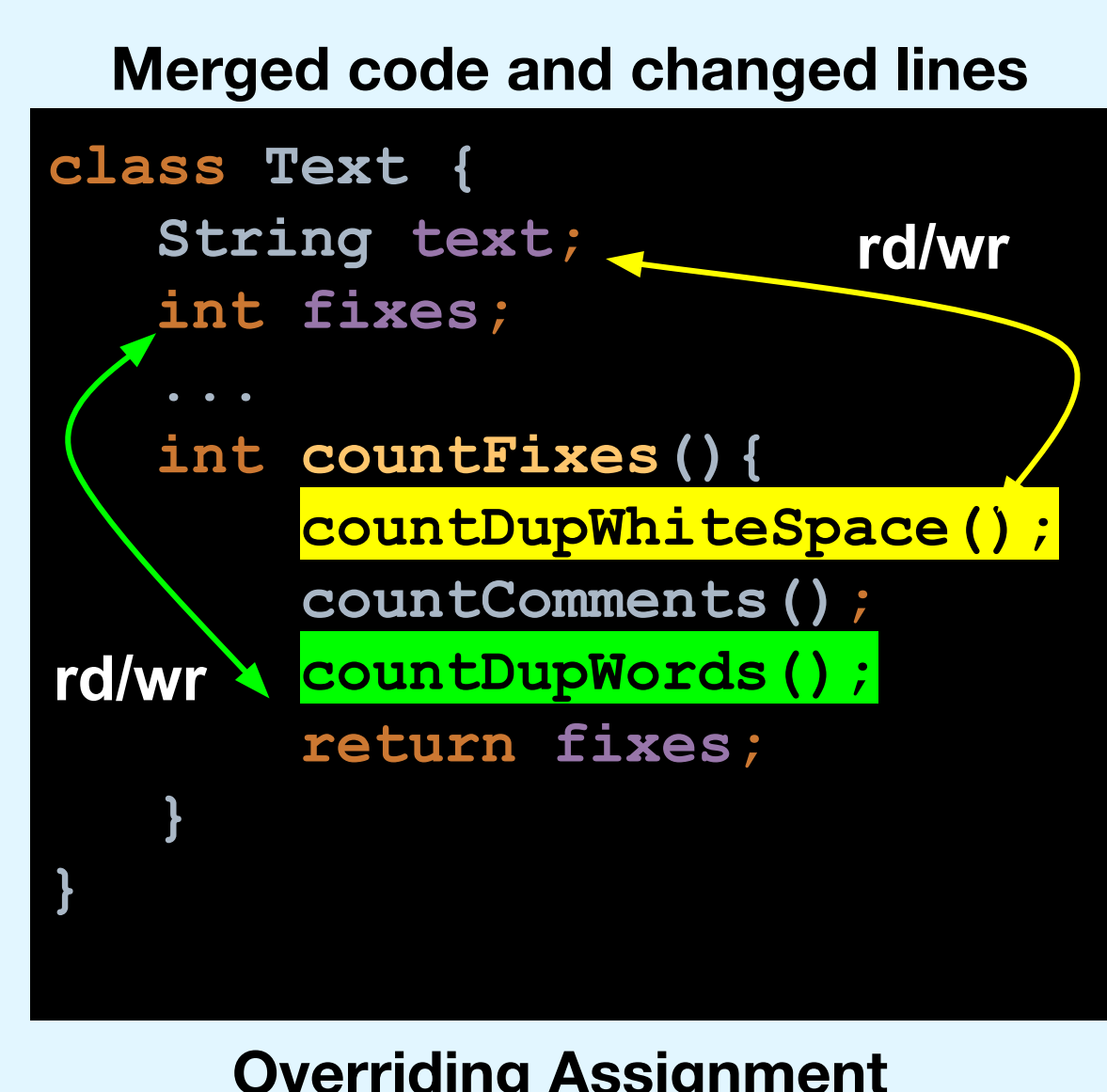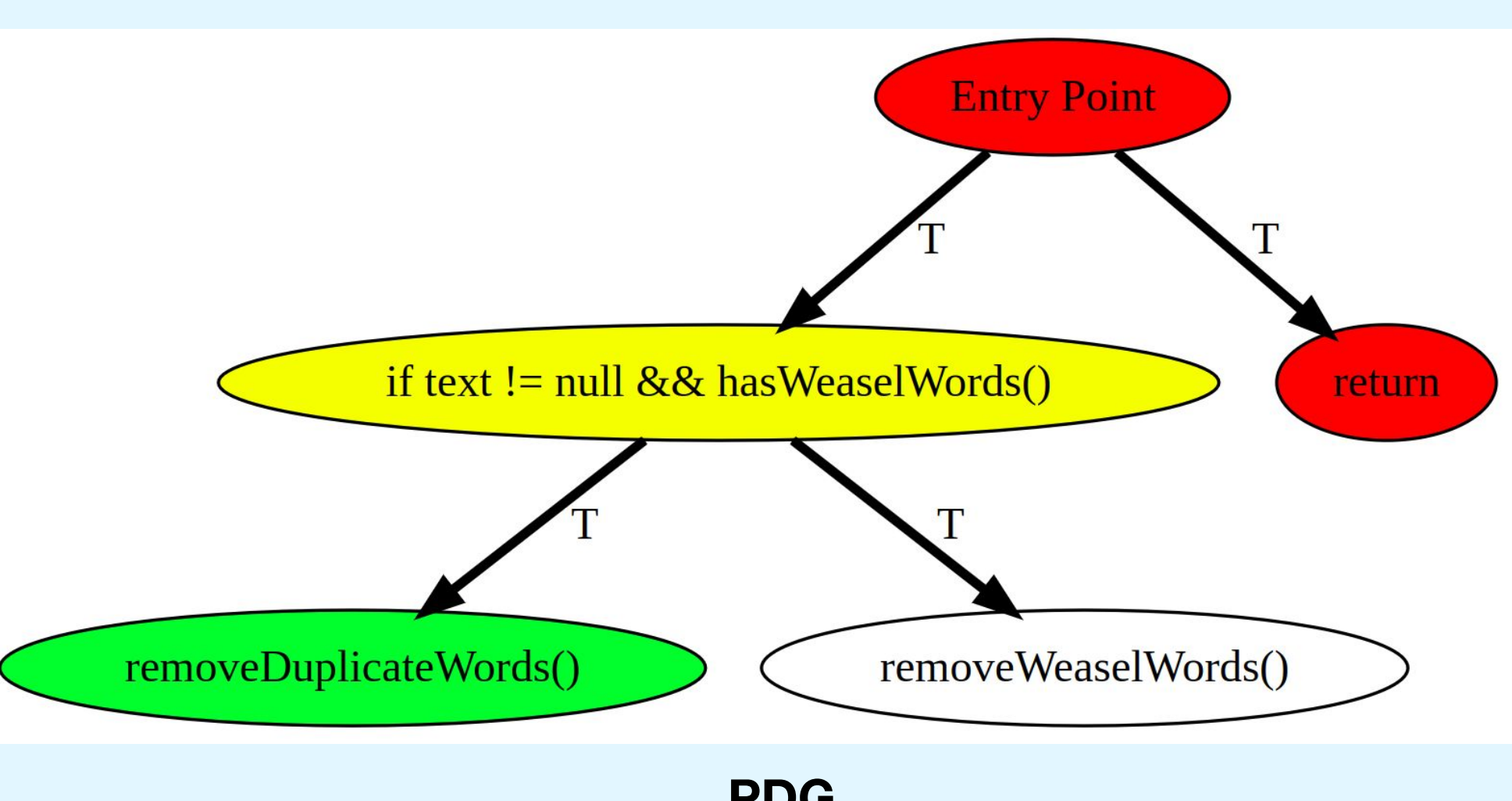Rodrigo Bonifácio[2]   Matheus Barbosa de Oliveira[1]

[1]Informatics Center, Federal University of Pernambuco, Brazil
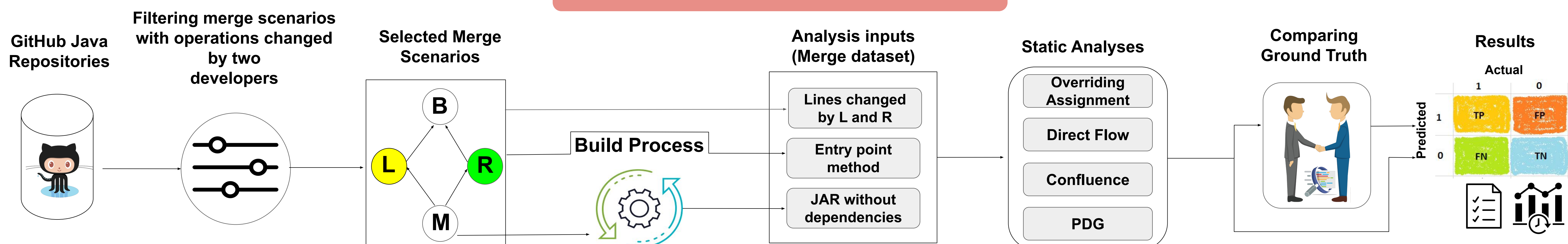[2]Computer Science Department, University of Brasília, Brazil
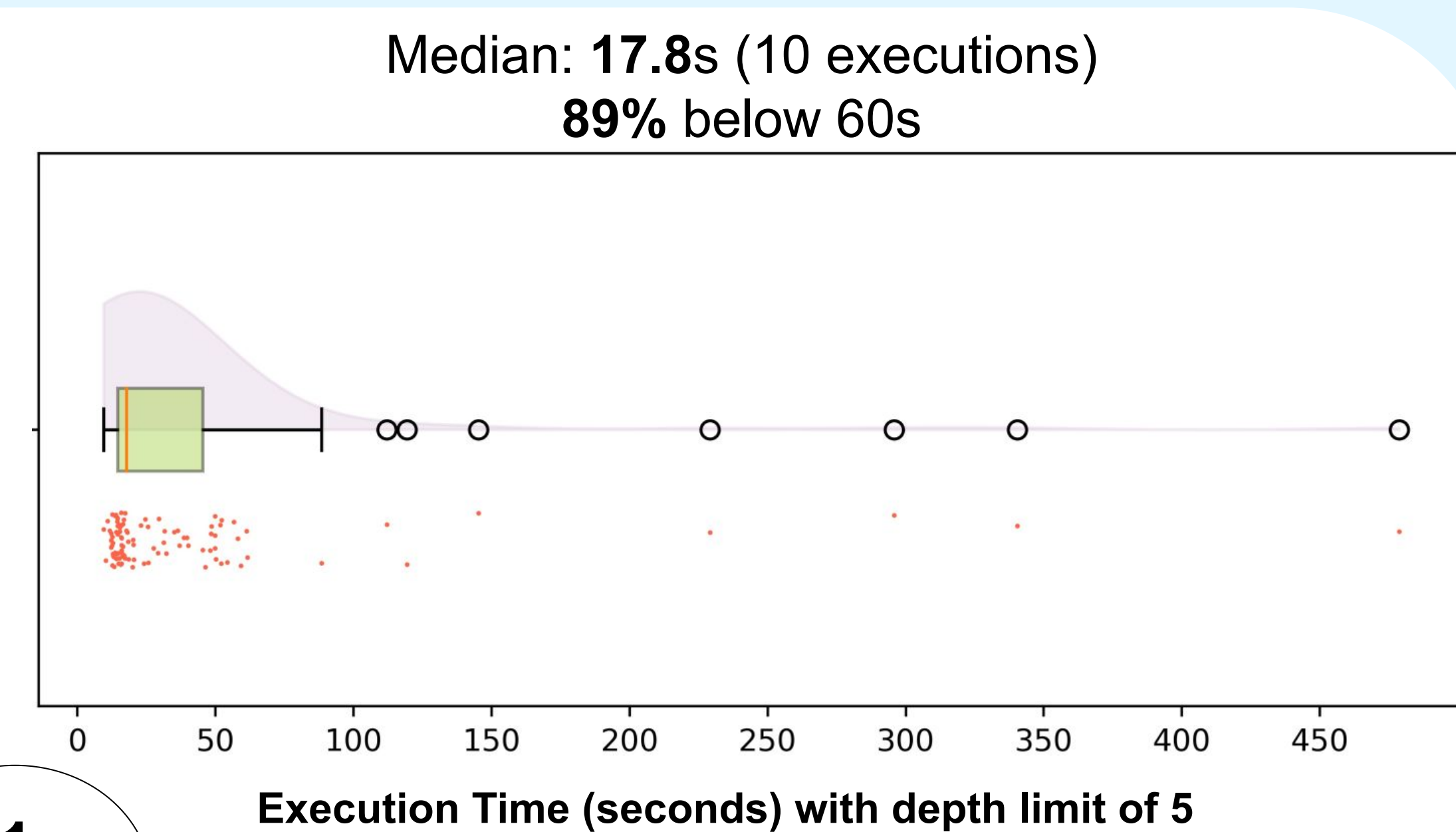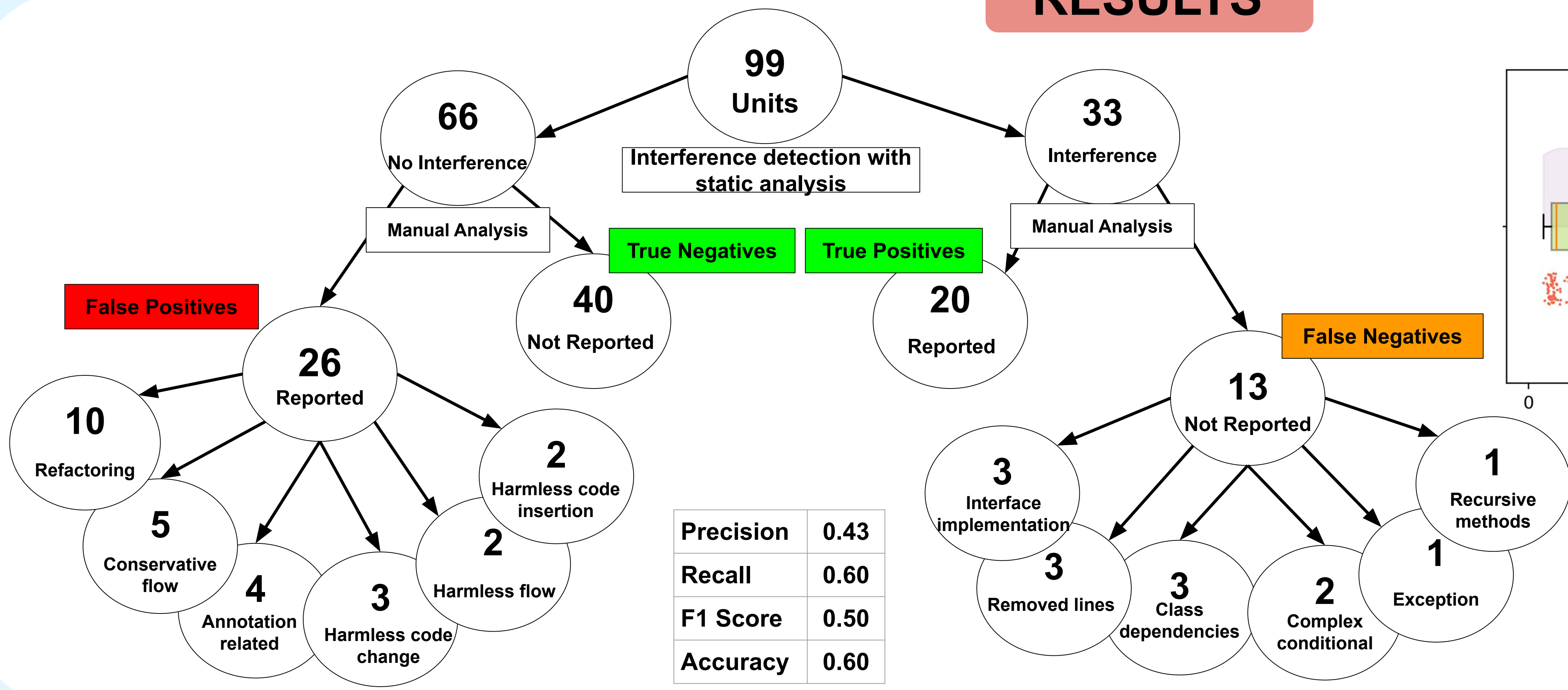
PROBLEM: INTERFERING CHANGES

Input: "the the dog"

BASE
```
1. cleanText(){
2.   if (text != null){
3.     removeWeaselWords();
4.   }
5. }
```
"the the dog"

LEFT
```
1. cleanText(){
2.   if (text != null
3.     && hasWeaselWords()){
4.   removeWeaselWords();
5.   }
6. }
```
"the the dog"

RIGHT
```
1. cleanText(){
2.   if (text != null){
3.     removeWeaselWords();
4.     removeDuplicateWords();
5.   }
6. }
```
"the dog"

MERGE
```
1. cleanText(){
2.   if (text != null
3.     && hasWeaselWords()){
4.   removeWeaselWords();
5.   removeDuplicateWords();
6.   }
7. }
```
"the the dog"

Left **interferes** with right

Behavior of the integrated changes does not preserve the intended behavior of the individual changes

SOLUTION

PDG

Merged code and changed lines
```
class Text {
  String text;
  int fixes;
  ...
  int countFixes(){
    countDupWhiteSpace();
    countComments();
    countDupWords();
    return fixes;
  }
}
```
Overriding Assignment

Merged code and changed lines
```
class Text {
  String text;
  int words;
  int spaces;

  int countSpaceWords(){
    countDupWhiteSpace();
    countComments();
    countWords();
    return spaces + words;
  }
}
```
Confluence

Merged code and changed lines
```
class Text{
  String text;
  ...
  void cleanText(){
    normalizeWhiteSpace();
    removeComments();
    removeDuplicateWords();
  }
}
```
Direct Flow

EMPIRICAL ASSESSMENT

RESULTS

99 Units — Interference detection with static analysis

66 No Interference → Manual Analysis
33 Interference → Manual Analysis

True Negatives — 40 Not Reported
True Positives — 20 Reported

False Positives — 26 Reported:
- 10 Refactoring
- 5 Conservative flow
- 4 Annotation related
- 3 Harmless code change
- 2 Harmless flow
- 2 Harmless code insertion

False Negatives — 13 Not Reported:
- 3 Interface implementation
- 3 Removed lines
- 3 Class dependencies
- 2 Complex conditional
- 1 Recursive methods
- 1 Exception

Precision	0.43
Recall	0.60
F1 Score	0.50
Accuracy	0.60

Median: **17.8**s (10 executions)
89% below 60s

Execution Time (seconds) with depth limit of 5

Depth limit of 5 on Ubuntu running on an Intel Xeon Gold 6338 server with 2TB of RAM