An Experience Report on Analyzing Industrial Software Systems Using Code Clone Detection Techniques

Norihiro Yoshida (NAIST)
Yoshiki Higo, Shinji Kusumoto, Katsuro Inoue (Osaka University)

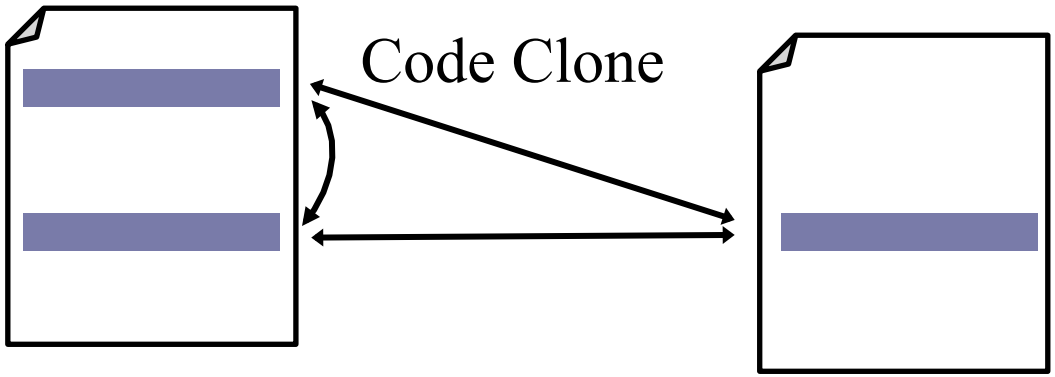
Outline

1. What is a code clone?
2. Discussions on the harmfulness of code clone
3. Importance of sharing industrial experiences with clone
4. Industrial application of clone analysis
   - Analysis tools
   - Result
5. Summary

What is a code clone?

- A code fragment that has identical or similar code fragments to it in source code.

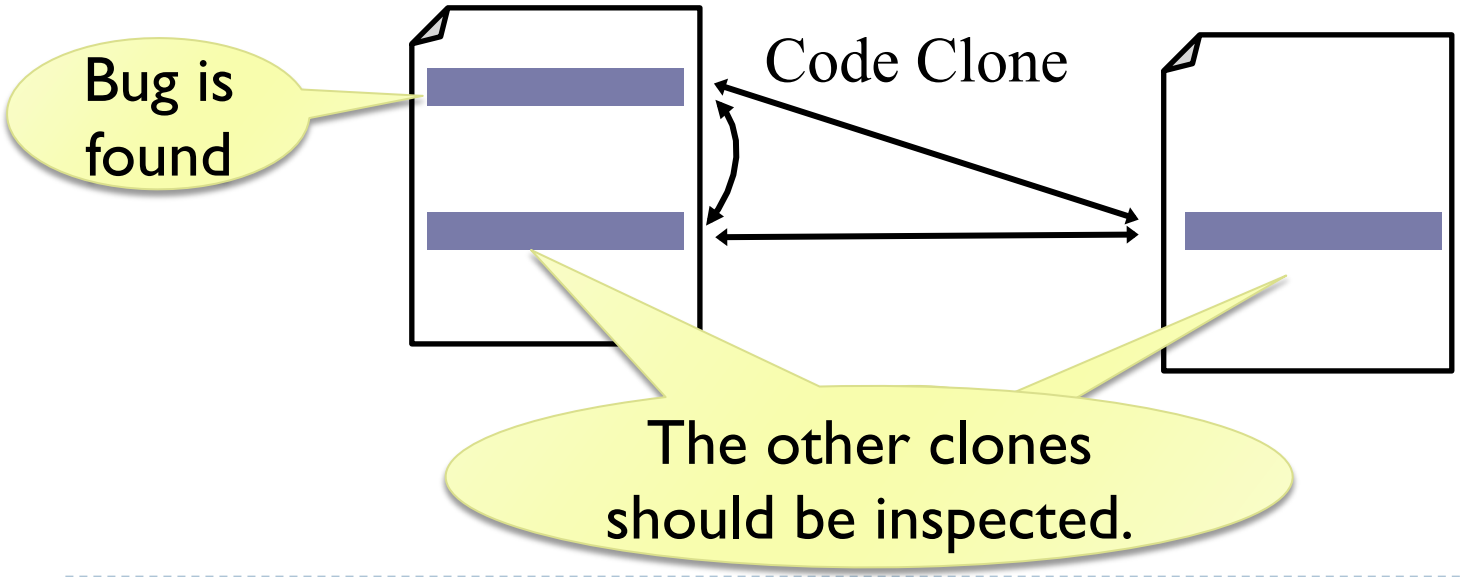
- Introduced in source program by various reasons such as reusing code by `copy-and-paste'



Discussions on the harmfulness of code clone (Opponent)

There have been numerous discussions:

- Cloning opponent: "***clone should be avoided because it makes software maintenance difficult.***"
  - Book on programming practice
  - Research papers that is a little bit less than state-of-the-art



Discussions on the harmfulness of code clone (Moderate & Proponent)

- Moderate: ***"Clone is unavoidable when a language lacks suitable modularization mechanism to eliminate it."***
  - e.g., it is difficult to merge code clones into a function in the case that the identifiers are unmatched.

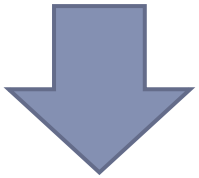
- Clone proponent: ***"Clones do not often cause bugs, so leave it be."***
  - According to Rahman's study [1] of OSS, there is no significant relationship among locations of bug and clones.

[1] Rahman, et al., "Clones: What is that Smell?", MSR 2011

Importance of sharing experiences with cloning

- Which is truth of code clones?
  - There is no conclusion currently.
  - Probably it depends on the context of cloning [2].

- Sharing experience with cloning is a promising way for easy identification of harmful and harmless clones.



- Software engineering community has to report experience with clone detection and analysis.

[2] Kapser, et al., ""Cloning considered harmful" considered harmful: patterns of cloning in software", Empirical Software Engineering, 2008.

Researches on cloning in industry

- Much research have been done on
  - Automatic clone detection
  - Analysis of code clones in OSS

- On the other hand, quantity of reports on cloning in industry has been lacking.
  - Ratio of clones to whole source code is higher in industry than in OSS
  - Rather than in OSS, clone causes a problem in industry.

It is needed to report an industrial experience with clone analysis.

Overview of industrial case study

1. Investigated an industrial software in terms of the following points by clone analysis technique.

    A. Is there significant difference in clones between the ends of the unit testing and the combined testing?
    B. Where clones are concentrated in the source code?
    C. What sort of characteristic clones are involved in the source code?

2. Interviewed developers for detected clones

Target software project

- Japanese governmental project
  - Software system for traffic infrastructure
- Source code
  - Approximately 100,000 LOC, and increased by 20 thousands after the unit test.
  - Main language is C/C++
- Organization
  - 5 vendors, each of which was assigned for a subsystem.
  - 1 project manager from a company different from the vendors

Tools for clone detection & analysis

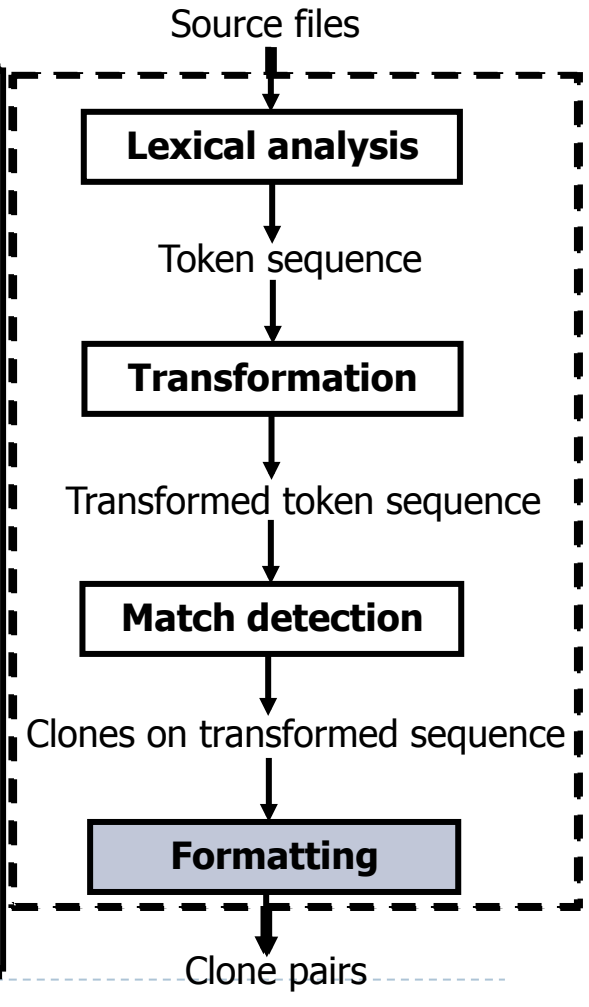
- Clone detection tool : CCFinder [3]
  - Detection of lexically-similar code clones based on the identification of identical token sequences in source code
- Code clone analyzer : Gemini [4]
  - Scatter plot
  - Metrics for extracting clones

[3] T. Kamiya, et al.: "A multilinguistic token-based code clone detection system for large scale source code", IEEE TSE, 2002.
[4] Y. Ueda, et al.: "Gemini: Maintenance Support Environment Based on Code Clone Analysis", METRICS 2002.

Token-based clone detection tool : CCFinder

Detection of identical token sequences in source code

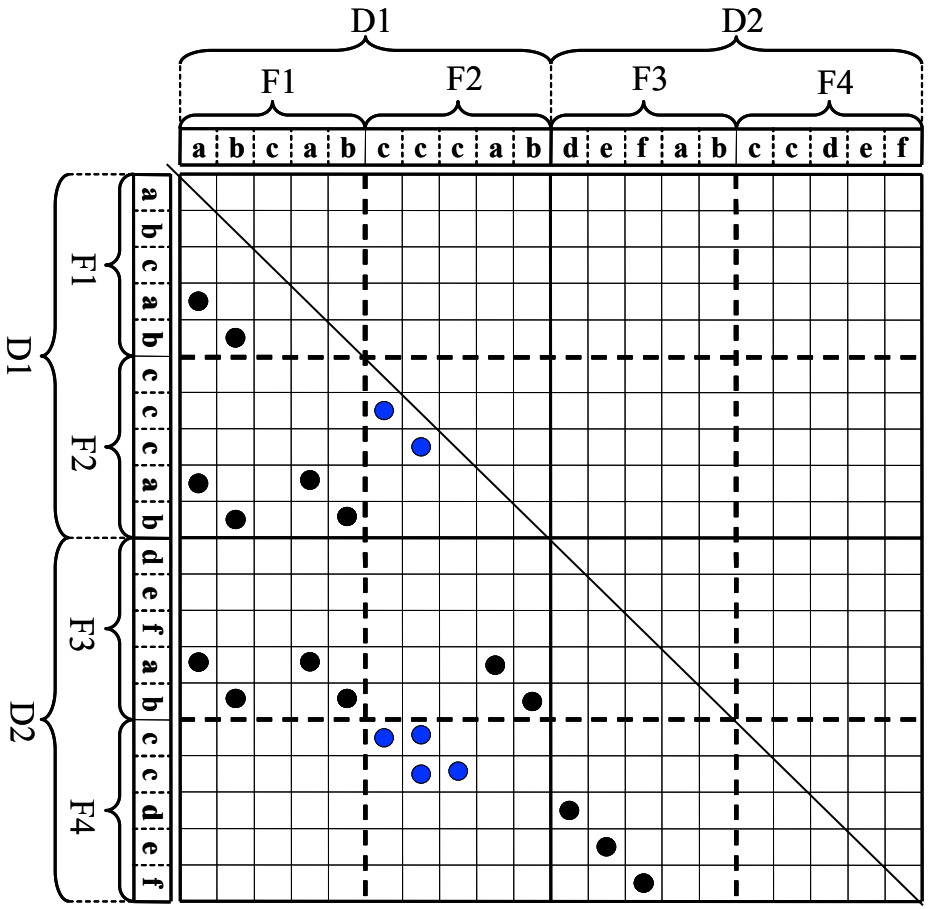
```
1. static void foo() throws RESyntaxException {
2.    String a[] = new String [] { "123,400", "abc", "orange 100" };
3.    org.apache.regexp.RE pat = new org.apache.regexp.RE("[0-9,]+");
4.    int sum = 0;
5.    for (int i = 0; i < a.length; ++i)
6.       if (pat.match(a[i]))
7.          sum += Sample.parseNumber(pat.getParen(0));
8.    System.out.println("sum = " + sum);
9. }
10. static void goo(String [] a) throws RESyntaxException {
11.    RE exp = new RE("[0-9,]+");
12.    int sum = 0;
13.    for (int i = 0; i < a.length; ++i)
14.       if (exp.match(a[i]))
15.          sum += parseNumber(exp.getParen(0));
16.    System.out.println("sum = " + sum);
17. }
```



Code clone analyzer : Gemini

Scatter Plot

- Visually shows where code clones are
- Both the vertical and horizontal axes represent the token sequence of source code
  - The original point is the upper left corner
- ● means that corresponding two tokens on the two axes are the same



Code clone analyzer : Gemini

Clone/File Metrics

- Example of clone metrics
  - ***LEN(S):*** the average length of code fragments (the number of tokens) in clone set ***S***
    - clone set : a set of code fragments, in which any pair of the code fragments is a code clone
  - ***NIF(S):*** the number of source files including any fragments of ***S***

- Example of file metrics
  - ***ROC(F):*** the ratio of duplication of file ***F***
    - if completely duplicated, the value is 1.0
    - if not duplicated at all, the value is 0.0
  - ***NOC(F):*** the number of code fragments of any clone set in file ***F***

Amount of Code Clones in Subsystems

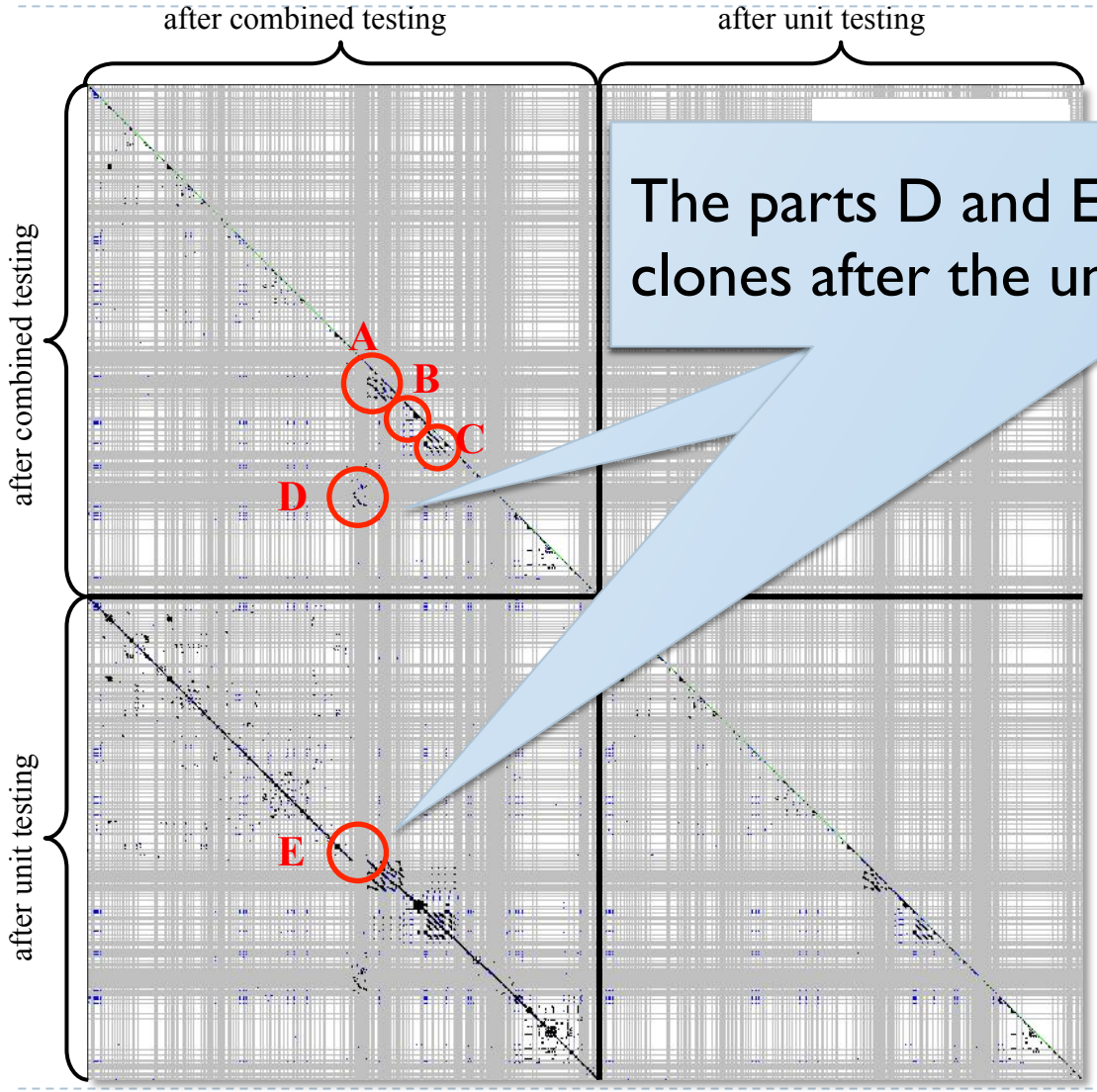
Company ID	After unit testing		After combined testing	
	# clones	Duplicated ratio	# clones	Duplicated ratio
V	259	34%	259	33%
W	369	27%	379	26%
X	4,483	55%	4,768	51%
Y	6,747	43%	7,628	46%
Z	2,450	56%	2,505	56%

Amount of Code Clones in Subsystems

Company ID	After unit testing		After combined testing	
	# clones	Duplicated ratio	# clones	Duplicated ratio
V	259	34%	259	33%
W			379	26%
X			4,768	51%
Y	6,747	43%	**7,628**	**46%**
Z	2,450	56%	2,505	56%

Clones had increased during combined testing

Scatter Plot (Company Y)

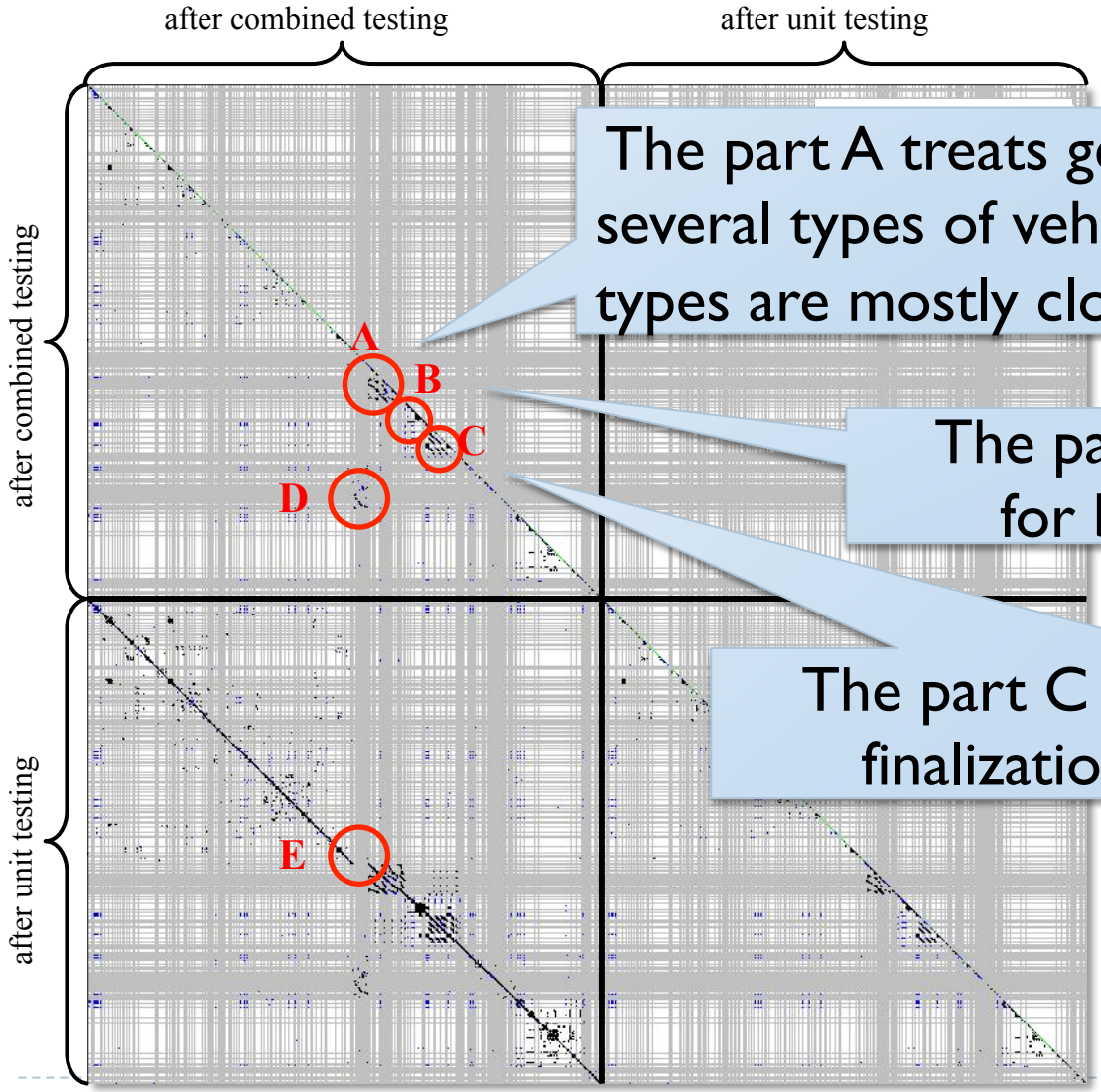


The parts D and E imply the creation of clones after the unit testing.

Interview

The developers insist that they added trusted library that has used in many products.

Scatter Plot (Company Y)



The part A treats geographical information of several types of vehicles. The code for the types are mostly cloned.

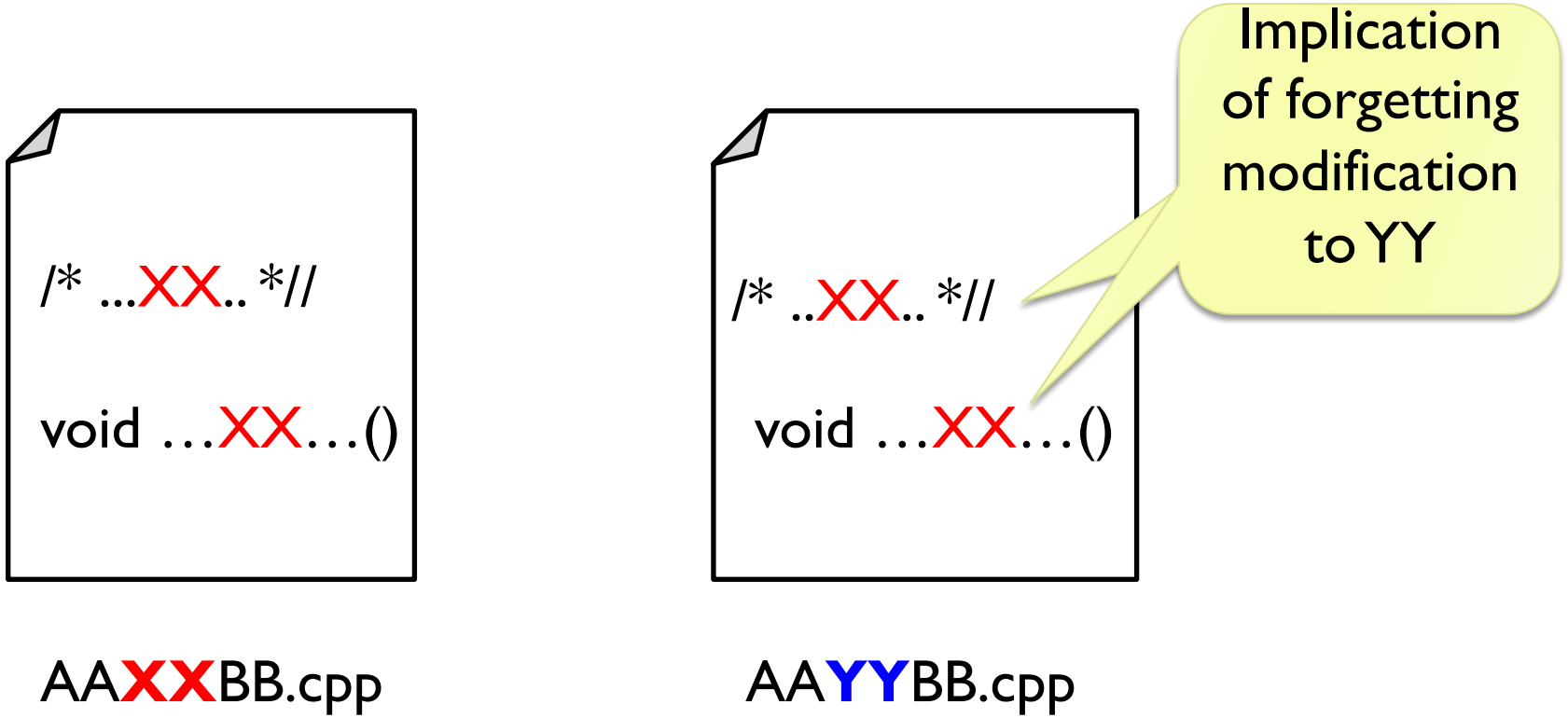
The part B involves statements for building SQL queries.

The part C involves initialization and finalization for a certain feature.

Example of detected clone

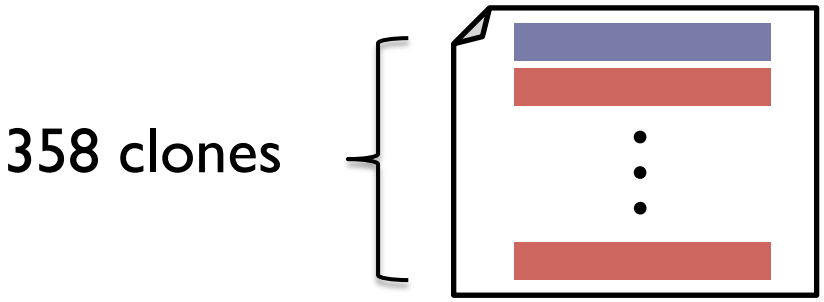
Clone metrics-based analysis

- Longest clones
  - A pair of 154 lines clones between the two files
  - Implications of copy-and-paste and forgetting modification

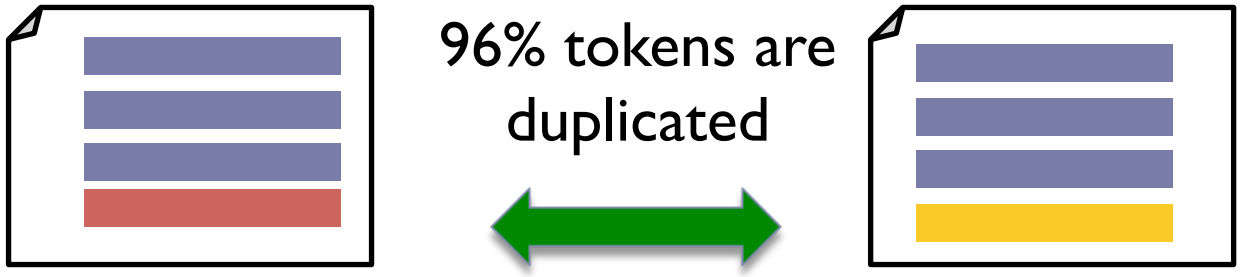


Example of detected clones
File metrics-based analysis

- Source file containing the maximum number of clones



- Most duplicated pair of source files



Interview

Developers had expected this duplication since design phase.

Summary & Future work

- Summary
  - Discussed the importance of sharing industrial experiences with clone analysis
  - Presented industrial application of clone analysis
    - Many characteristic clones were extracted
    - According to interviews for some of the extracted clones, the developers expected the existence of clones.

- Future work
  - Conduct the further analysis for determining whether harmful clones or not