API change extraction

Applicable for students as HiWi, Bachelor of Science, Master of Science
Keywords: *Source Analysis, Versioning, Software Quality*

Introduction

When developing and adding new features to libraries or programs, interoperability with existing code is one of the highest priorities. In some cases API-breaking changes are however unavoidable. For some libraries, these are described in version change-logs, but a developer needs to manually find them and adjust the program accordingly.

Tim Heldmann
tim.heldmann@tu-darmstadt.de

Office: S1|03 Room 4a
Hochschulstraße 1
64283 Darmstadt
Tel. 06151 16-27275

Date: 28th February, 2023

Task

Develop a tool, which compares two versions of a program or API, and detects changes. The versions are given either as separate folders, git commits, or tags. The changes we are most interested in are function declaration changes, and state knowledge transfers. Changes like scope-movement (movement to a different namespace, or class), are a nice to have. The tool will output the changes in a JSON structured description format, to be used in other refactoring tools.

```
1  int lib_func(int a, int b);
2  //changed to
3  double lib_func(int a, int b, bool useUnsafeMath);
```

Figure 1: A library function that had a change in its return value, and signature

```
1  node.getChildren();
2  //changed to
3  graph.getChildren(node)
```

Figure 2: The knowledge of program state has been transferred to a graph object

What you will be doing

(a) Develop a Tool, which detects API breaking changes in function signatures, states or similar

(b) Analyze the difficulties in scope-movement detection, and implement a draft

(c) Evaluate w.r.t. applicability and commonness of occurrence

Qualifications

- Familiarity with Git[2], C/C++,

- Experience in either source-code or text analysis.

- Some first experiences with clang[1] as a syntax analyzer

References

[1] https://clang.llvm.org/docs/LibTooling.html

[2] https://git-scm.com/