Eötvös Loránd University, Faculty of Informatics

Incremental parsing of build systems

Máté Cserép[1] **Anett Fekete**[2]

[1]Department of Software Technology and Methodology [2]Department of Programming Languages and Compilers

Introduction

The maintenance of large software results in higher development time and cost due to increasing size and complexity of the codebase and its documentation, their continuously eroding quality and fluctuation among developers. The main task of a code comprehension software tool is to provide exact textual information and visualization views regarding the analyzed codebase to support the understanding of the source code. For an enterprise software under development, this requires the frequent static reanalysis of the program. Performing a complete analysis each time is a significant waste of computational resources and could take several hours for a large software.

To reduce the time and cost of parsing, we introduced incremental parsing that relies on the abstract syntax tree of the code, the defined direct dependencies between files (e.g. *header inclusions*) and on the changes of the build instructions.

CodeCompass

CodeCompass is an open source code comprehension software developed by Ericsson Ltd. in cooperation with Eötvös Loránd University. It provides various navigational functions and textual search about the processed code by not only relying on the codebase but by obtaining information from the build system. The latter is done by processing the compilation commands generated for the project (e.g. with *CMake*) or logged otherwise during build time.

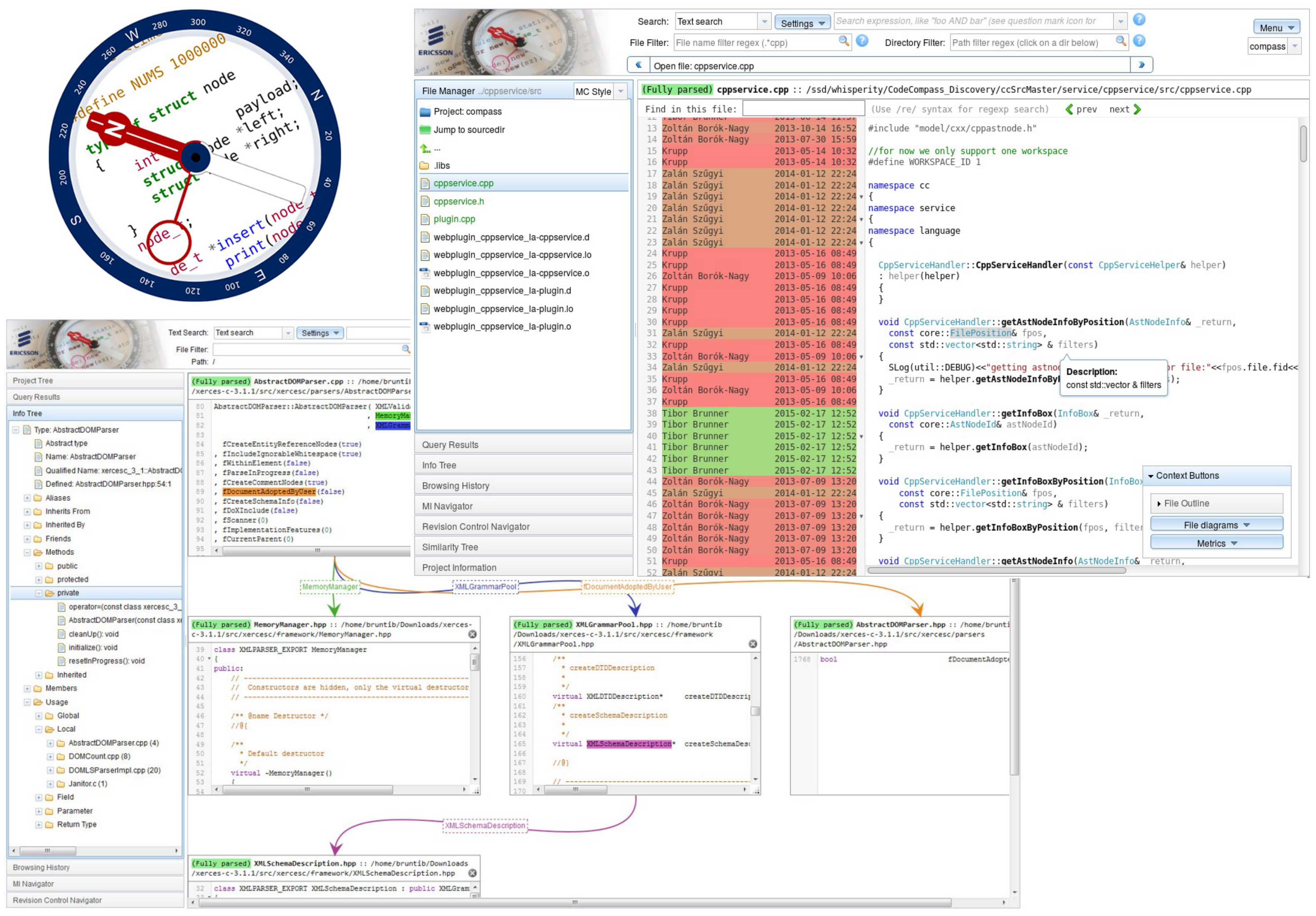

Figure 1: Web frontend of the CodeCompass code comprehension tool.

Live demo

Online demonstration of CodeCompass, showcased on the *LLVM* project is available at: `https://codecompass.zolix.hu/`

Incremental parsing workflow

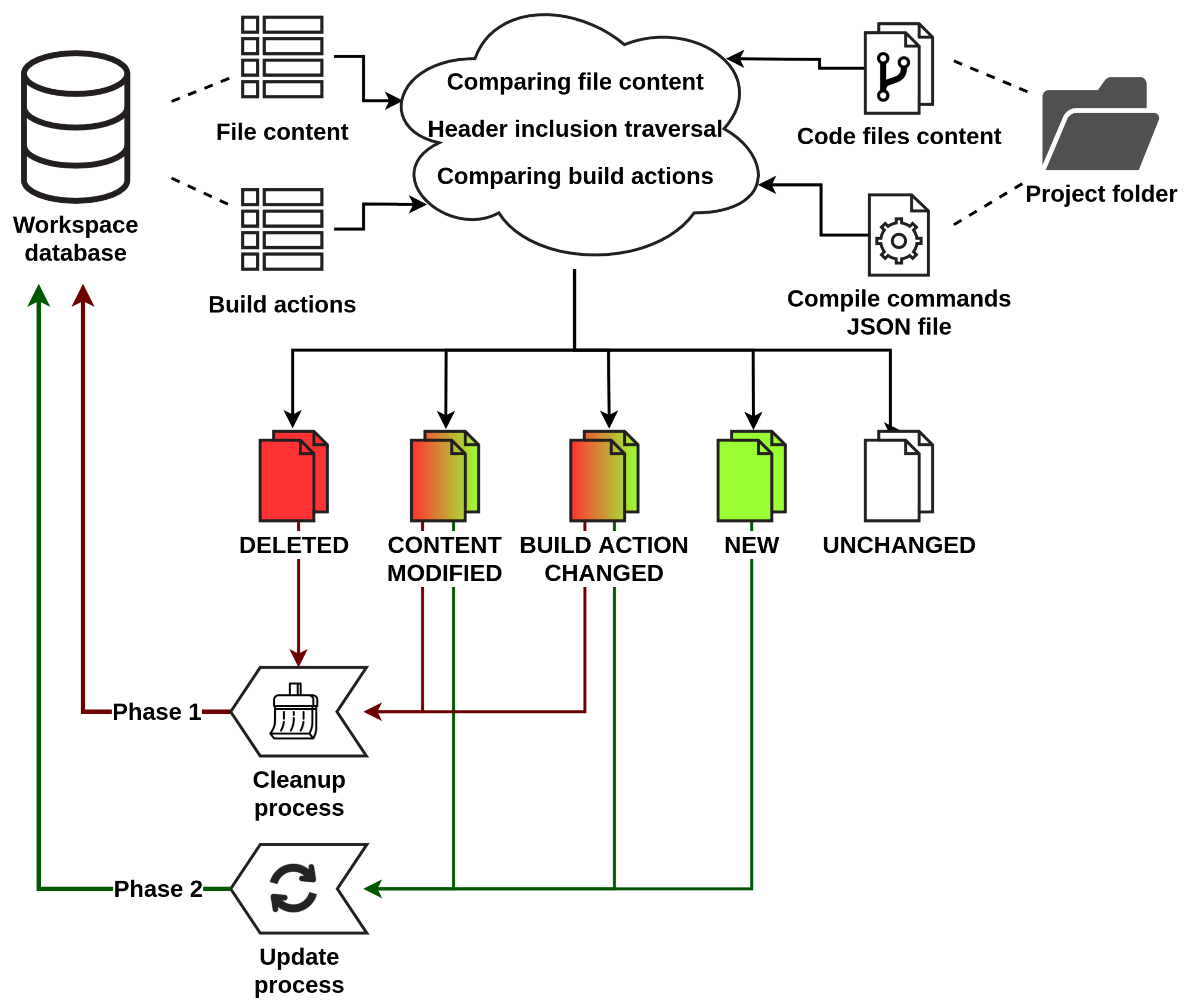

Figure 2: Workspace database update workflow during incremental parsing.

Results

A performance test of incremental parsing was evaluated on the open-source *LLVM* project. LLVM is a large C++ software containing nearly 5000 C++ source files, and is actively developed with a frequently changing build system. The testing environment was an average personal computer with 4 CPU cores and 16GB RAM.

Test case	Files	CPU time	Wall time
Clean build	all	387 min	127 min
Incremental build #1	1	89 sec	70 sec
Incremental build #2	7	110 sec	95 sec
Incremental build #3	17	169 sec	206 sec
Incremental build #4	853	160 min	119 min

Table 1: Comparison of a full and various incremental parsing cases on LLVM.

References

[1] Z. Porkoláb, T. Brunner, D. Krupp, M. Csordás: CodeCompass: An open software comprehension framework for industrial usage. *In Proceedings of the 26th Conference on Program Comprehension.* ICPC '18. pp. 361–369., 2018

[2] A. Fekete, M. Cserép: Incremental Parsing of Large Legacy C/C++ Software. *In 21th International Multiconference on Information Society* vol. G, pp. 51–54., 2018

ERICSSON

Acknowledgement

EFOP-3.6.3-VEKOP-16-2017-00002:
This work is supported by the European Union, co-financed by the European Social Fund.

Information

Authors: Máté Cserép, Anett Fekete
Contact: {mcserep, afekete}@inf.elte.hu
Date: 2020 January 24