

# Evaluating Comment-to-AST Assignment Heuristics for C++ Programs

Tamás Cséri, Zoltán Porkoláb

Department of Programming Languages and Compilers  
Eötvös Loránd University  
{cseri,gsd}@caesar.elte.hu

Comments are integral part of the source code of software. They preserve the intentions of the developers, document constraints and highlight implementation details. Good comments help us to understand the codebase and make maintenance easier. Most of the software tools ignore comments because they take no part in code generation. However, there are cases when comments should be taken into account: refactoring tools need to move code along with their comments [1] and code comprehension tools need to show comments related to a given context [2]. Since these tools are working on the AST, comments should be assigned to the appropriate AST nodes.

Assigning comments to AST nodes is a non-straightforward task. Most methods use heuristics that place the comment to the proper AST node. This article improves existing heuristics. We identify corresponding AST nodes by distance and type. We also manage to contract consecutive connected comments. Macro-related comments are handled in a special way.

We quantify the correctness of comment assignments and evaluate the different solutions on open source C++ projects comparing our method with existing tools. Our method may be useful for other programming languages with respective modifications.

## References

- [1] P. Sommerlad, G. Zraggen, T. Corbat, and L. Felber. Retaining comments when refactoring code. In *Companion to the 23rd ACM SIGPLAN conference on Object-oriented programming systems languages and applications*, OOPSLA Companion '08, pages 653–662, New York, NY, USA, 2008. ACM.
- [2] T. Cséri, Z. Szúgyi, and Z. Porkoláb. Rule-based Assignment of Comments to AST Nodes in C++ programs. In *Proceedings of the Fifth Balkan Conference in Informatics*, BCI '12, pages 291–294, New York, NY, USA, 2012. ACM.