# Compiler Front End Fusion: Undo Desugaring in Language Processing Tools

## Artúr Poór, Tamás Kozsik, Melinda Tóth, István Bozó

Department of Programming Languages and Compilers
Eötvös Loránd University, Budapest, Hungary

{poor_a, kto, toth_m, bozo_i}@inf.elte.hu

Compiler front ends often perform *desugaring* on the source code while constructing the abstract syntax tree (AST). A programming language processing tool (such as a refactoring tool) working with the desugared AST perceives the code at this abstract level, and loses information on the rich syntax used in the actual source code. This paper discusses the concept of *front end fusion*, a technique which may help language processing tools to retain the syntactic sugar information on the source code in the presence of desugaring compiler front ends. We propose a hybrid front end created from two separate front ends: one provided by the compiler, which offers type information, and another one, which provides the details of the concrete syntax used in the source code. As a case study, we show how to construct a hybrid front end in a language processing tool for the Scala programming language.

# References

[1] Artúr Poór, István Bozó, Tamás Kozsik. Resugaring: Undo desugaring in language processing tools. Thessaloniki, Greece, 2017. To appear in the *Proceedings of the Symphosium of Computer Languages and Tools*.

[2] Huiqing Li and Simon Thompson. *Tool support for refactoring functional programs*. In Partial Evaluation and Program Manipulation, San Francisco, California, USA, January 2008. Assoc of Computing Machinery.

[3] Dean Wampler and Alex Payne. *Programming Scala – Scalability = Functional Programming + Objects*. OReilly Media, 2nd edition, December 2014.