

# Data-Type-Generator for Bolare

Siegfried Jurschitz

TU Wien

[e0425419@student.tuwien.ac.at](mailto:e0425419@student.tuwien.ac.at)

## Abstract

Rebol is a very powerful and easy-to-use programming language, that comes with a great number of built-in functions like network protocols and a vast amount of native data-types like ULRs, tupels, and many more. Even though this language comes with a rich package, it has some drawbacks: only one developer is currently developing new functionality or providing bug-fixes. Some known issues weren't resolved for several years. This closed-source project inspired a new open-source programming language, which is currently under development: Bolare. Bolare should ease young students into the highly complicated world of programming with its simple syntax and high level of comfort it provides. Like Rebol, Bolare should thrive on its rich content and data-types it is delivered with. In the current situation, it is rather difficult to implement new data-types, because each new type would result in a multitude of changes, beginning at recompiling the compiler itself. This issue is to be solved by introducing a data-type-generator.

At this point, Context Free Grammars (CFGs) and Parsing Expression Grammars (PEGs) come to notice, because either one should be used for parsing a data-type definition to provide Bolare with a rich set of native data-type-definitions as well as self-made data-types, that may be introduced at run-time.

To decide, which method is more suitable to get implemented for the use in Bolare, both advantages and disadvantages will be shown for the mentioned technologies.