



MUGLA SITKI KOCMAN UNIVERSITY

Implementation of Domain Specific Language Using Rust

Ali Dikme – Mehmet Reşit Çağan

Abstract

This project showcases the development of a Rust-based interpreter for a simplified programming language. The interpreter, constructed step-by-step, demonstrates the fundamental processes of lexical analysis, parsing, and code execution using abstract syntax trees (ASTs). From tokenization to the handling of variables and functions, the poster illustrates how Rust's language features empower the creation of a robust interpreter. Through practical examples and insights, this project illuminates the journey from language specification to functional interpreter, offering valuable lessons in systems programming and language implementation.

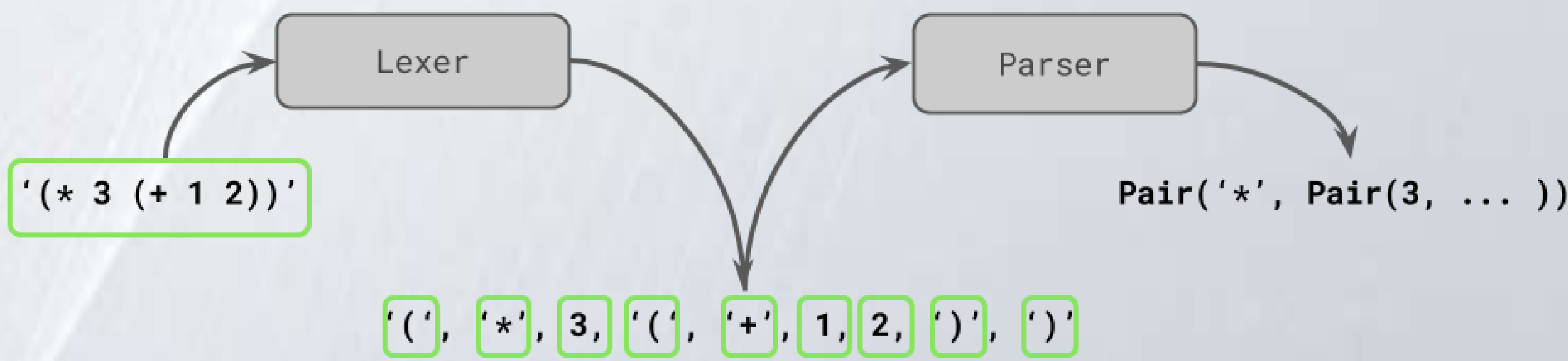
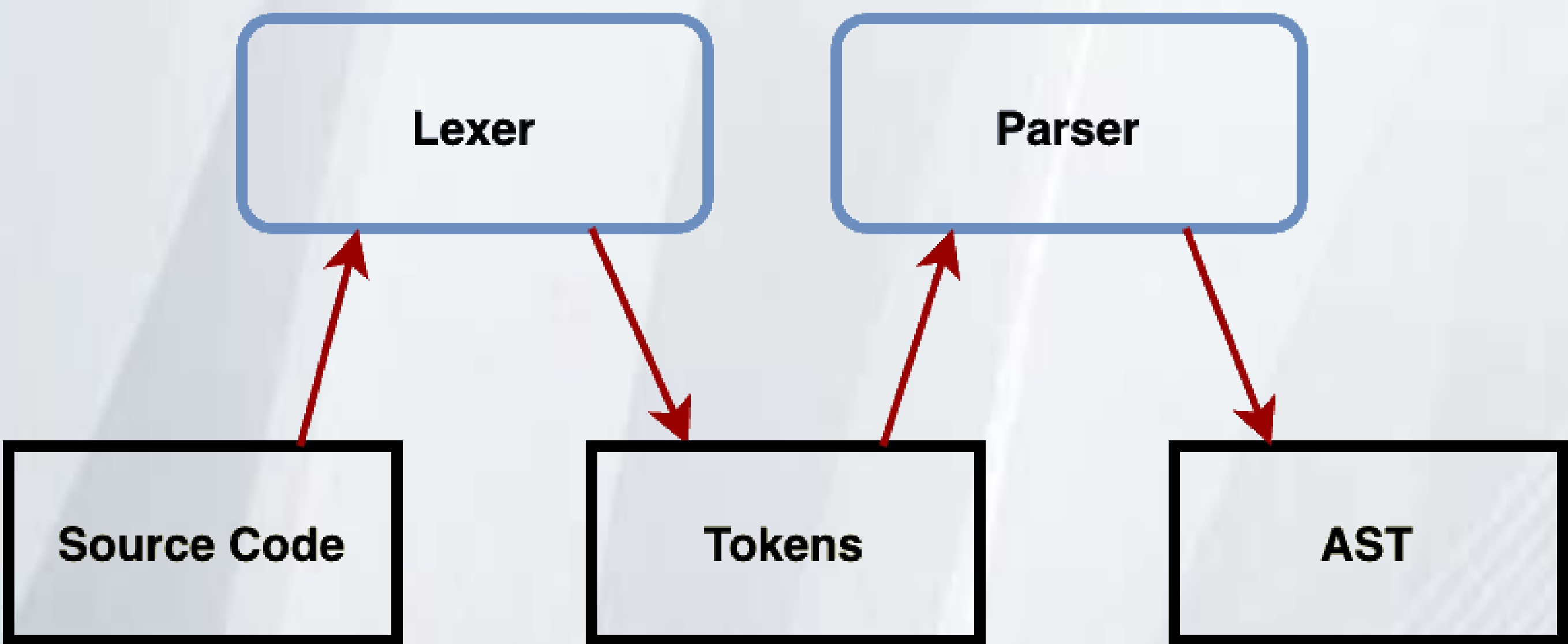
Language



**The Rust
Programming
Language**

Key Concepts

- Lexical Analysis (Lexing) and Tokenization
- Parsing Techniques and Abstract Syntax Trees (ASTs)
- Code Evaluation and Execution
- Management of Variable Assignments, Function Calls, and Control Structures
- Utilization of Rust Language Features for Language Implementation



Explore Rust's
unique features
and its rising
popularity

Creating a
specialized
language using
Rust

Emphasize the
excitement of
creating a new
language using
Rust