**Compiler-Compilers (Compiler/Parser Generators)**

Compiler-compilers are tools that are used to produce a parser from, essentially, a grammar.

**The Dinosaurs**

The most famous examples of such tools are Lex and Yacc.

* Lex is a lexical analyzer: it transforms a stream of characters into a stream of tokens. Valid tokens are described eg. with regular expressions.
* Yacc (Yet Another Compiler Compiler) is a parser generator: from a grammar-like description, it generates a parser with semantic actions

Usually present on Unix distributions nowadays are Flex (Faster, newer lex) and Bison, a backwards-compatible rewrite of Yacc. By force of habit, though, they’re still often referred to as Lex and Yacc.

These tools have been around for a while (since the 70s/80s), but are still very commonly used in C/C++ projects. From the Lex manual:

"The asteroid to kill this dinosaur is still in orbit."

**Compiler-Compilers for Java**

* [JFlex](http://jflex.de/) and [CUP](http://www2.cs.tum.edu/projects/cup/), written with the goal of resembling Lex and Yacc
* [JavaCC](https://javacc.dev.java.net/), Example: [JavaCC partial parser description of the while language](http://lara.epfl.ch/w/compilation:ex3-javacc-stub).
* [SableCC](http://www.sablecc.org/). Can (semi-)automatically generate a class hierarchy representing the syntax trees.
* [ANTLR](http://www.antlr.org/) (ANother Tool for Language Recognition)