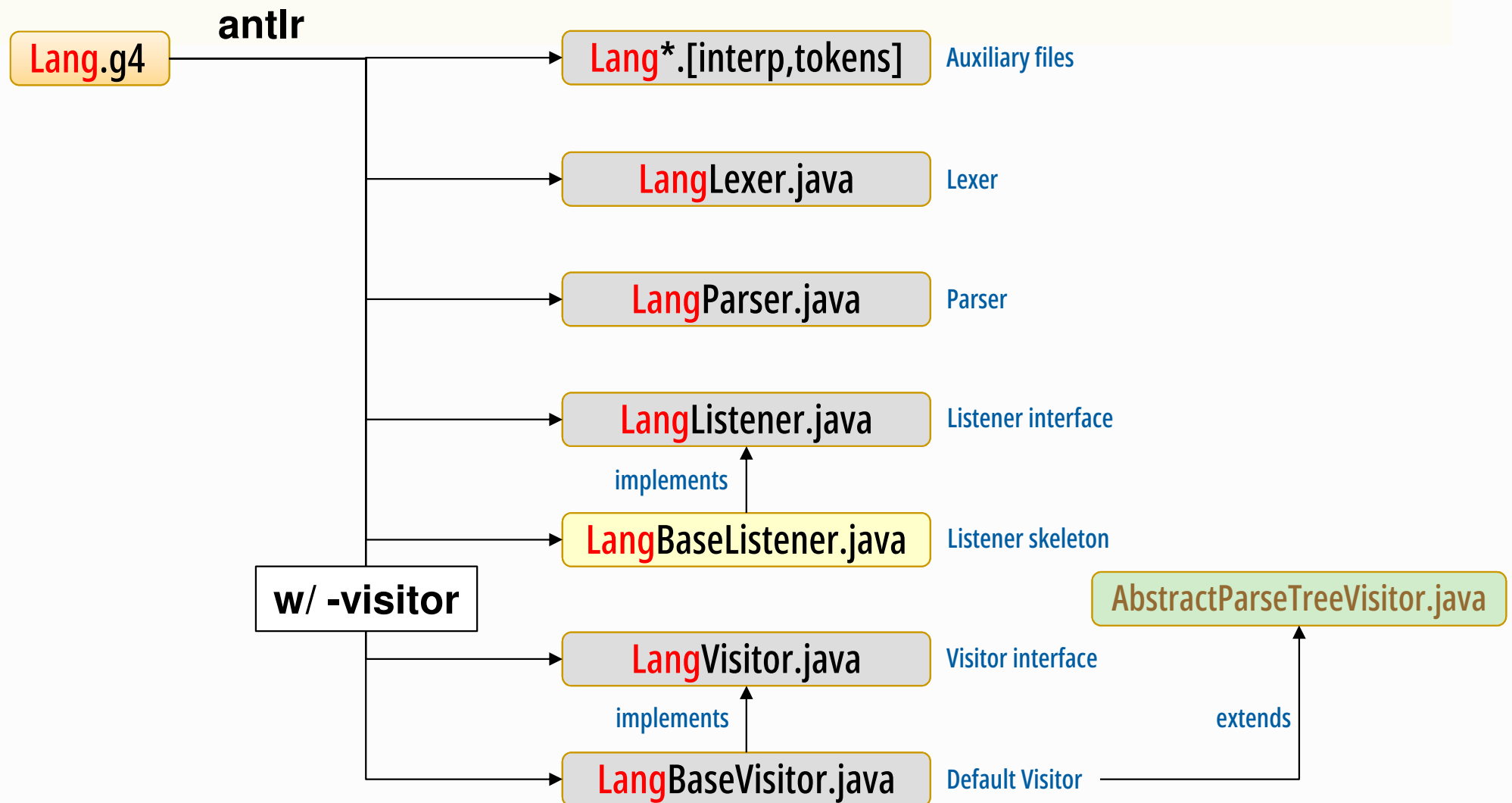


# ANTLR 4

# Automatic code generation



# Manual code generation

▷ To implement a compiler we develop one or more

## ♦ Listener

- Copy **BaseListener**
  - Edit the copy
  - New grammar rules → add new methods manually
    - a dummy listener can be created for that
- Alternative: **antlr4-listener**

MyListener.java

## ♦ Visitor

- Copy **BaseVisitor** or create from scratch
  - Extend only **BaseVisitor**
    - And define a specific return type for all methods
  - New grammar rules → add new methods manually
    - But on a needed basis (a dummy visitor can be created for that)
    - Default recursive methods are implemented by **BaseVisitor**
- Alternative: **antlr4-visitor**

MyVisitor.java

# Mixed code generation / editing

- ▶ To use ANTLR4 code we need a tedious setup

- ◆ Which is mainly automatic

- ▶ The **antlr-main** command does that

- ◆ Creates the main entry point for the compiler

- ```
antlr4-main Lang.g4 1st_rule [-v Visitor] [-l Listener]
```

- ◆ The file can (must) be edited

- The command refuses to overwrite it (unless using **-f**)

LangMain.java



- ▶ Other options

- i → line oriented interpreter

- ◆ The grammar may be omitted (if unique on the directory)

- ◆ The 1<sup>st</sup> rule must be omitted (the first of the grammar is assumed)