Teil II. Elementare Analysewerkzeuge

Prof. Dr. rer. nat. Uwe Aßmann
Institut für Software- und Multimediatechnik
Lehrstuhl Softwaretechnologie
Fakultät für Informatik
TU Dresden
http://st.inf.tu-dresden.de
Version 13-1.0, 05.12.13

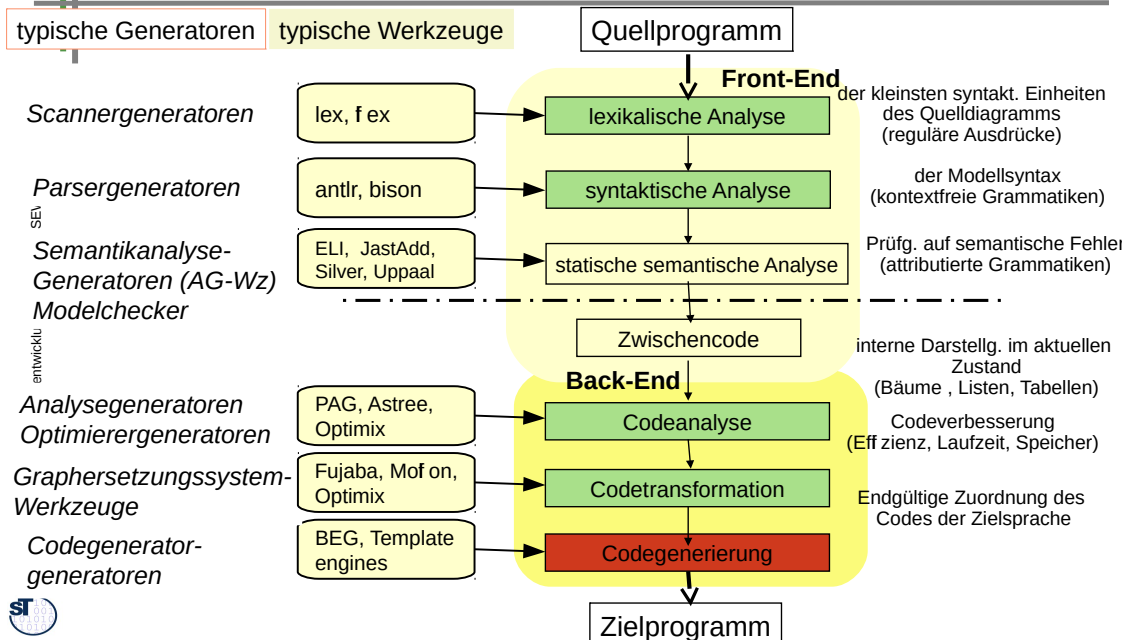
20. Parser-Generatoren im Technikraum Grammarware

1) Grundlagen
2) Beispiel Taschenrechner

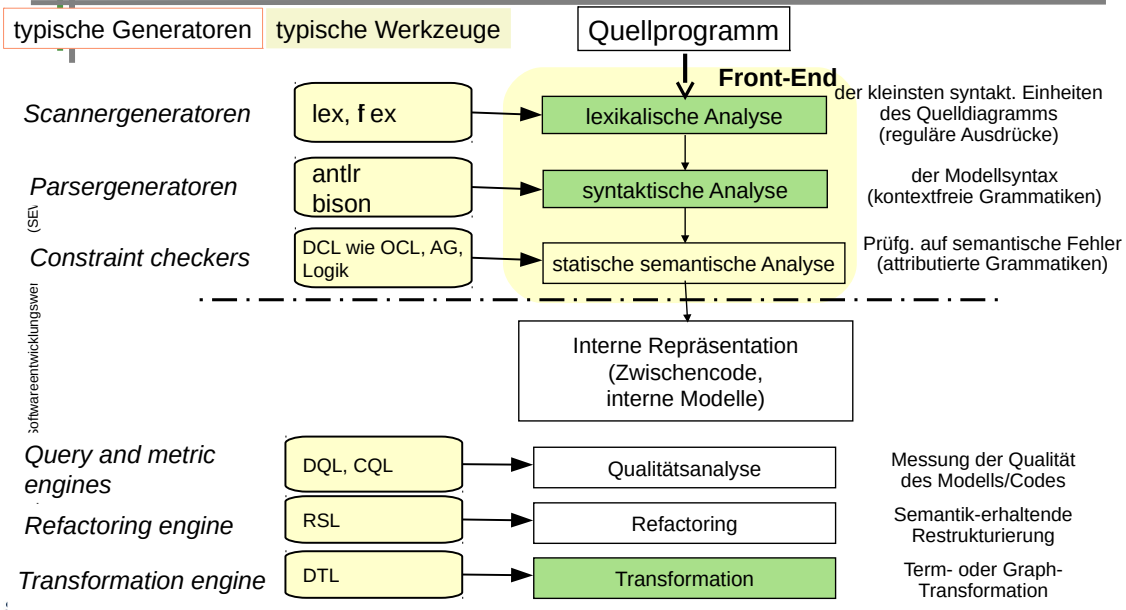
Literatur

- Obligatorisch:
- http://www.antlr.org
- Zusätzlich:
  - Cocktail www.cocolab.de, die Compiler-Toolbox für die schnellsten Compiler der Welt (kommerziell, Demoversionen erhältlich)

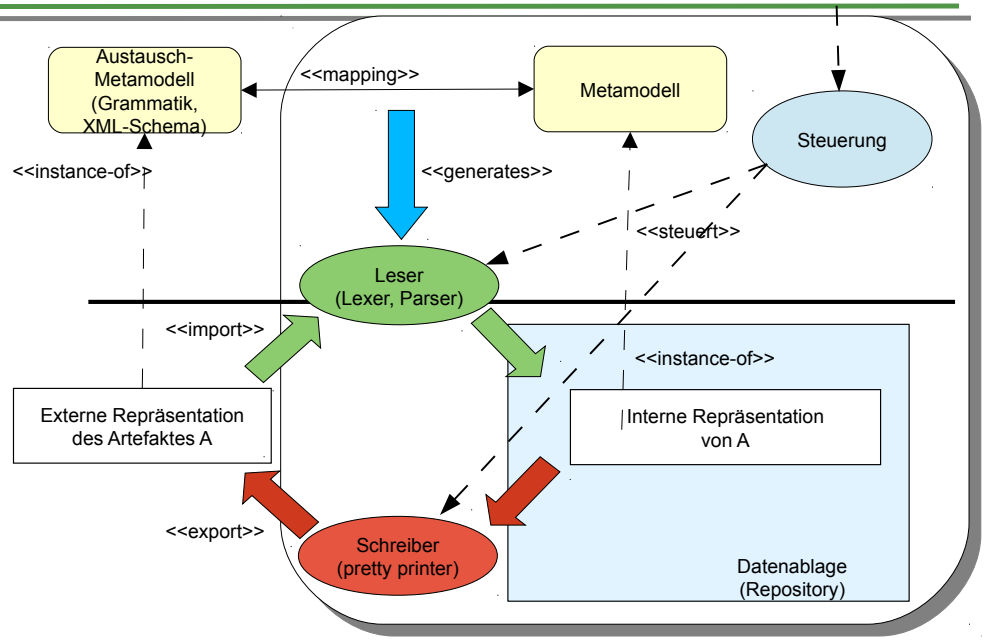
Phasen eines Werkzeugs und ihre erzeugenden Werkzeuge



5 Phasen eines Importers in ein Repository und die erzeugenden Werkzeuge



6 Wdh: Nutzung von generierten Parsern zum Import in Repositorien



7 Problem

- Parsen eines Programms, Modells oder Artefakts bedeutet, seine kontextfreie Syntax zu erkennen
- Parser sind die ersten Phasen eines Werkzeugs, denn es muss ein Artefakt importieren und damit ihn parsen
- Parsen erzeugt einen *Syntaxbaum*
- Parser wurden ursprünglich von Hand geschrieben (Compilerbau), heute generiert man sie aus *Grammatiken in EBNF*
  - **Parser** erkennt die Struktur des Textes anhand der Grammatik (“Zerteiler”)
  - **Baumaufbauer (Baumkonstruktor)** erzeugt einen abstrakten Syntaxbaum. Normalerweise wird er mit dem Parser verschränkt.

Wie arbeite ich flexibel mit mehreren Programmiersprachen oder DSL?

8 Antwort

- Bidirektionale Abbildung zwischen Technikraum “Grammarware” und einem anderen Technikraum, wie z.B. “Treeware” oder “Modelware”

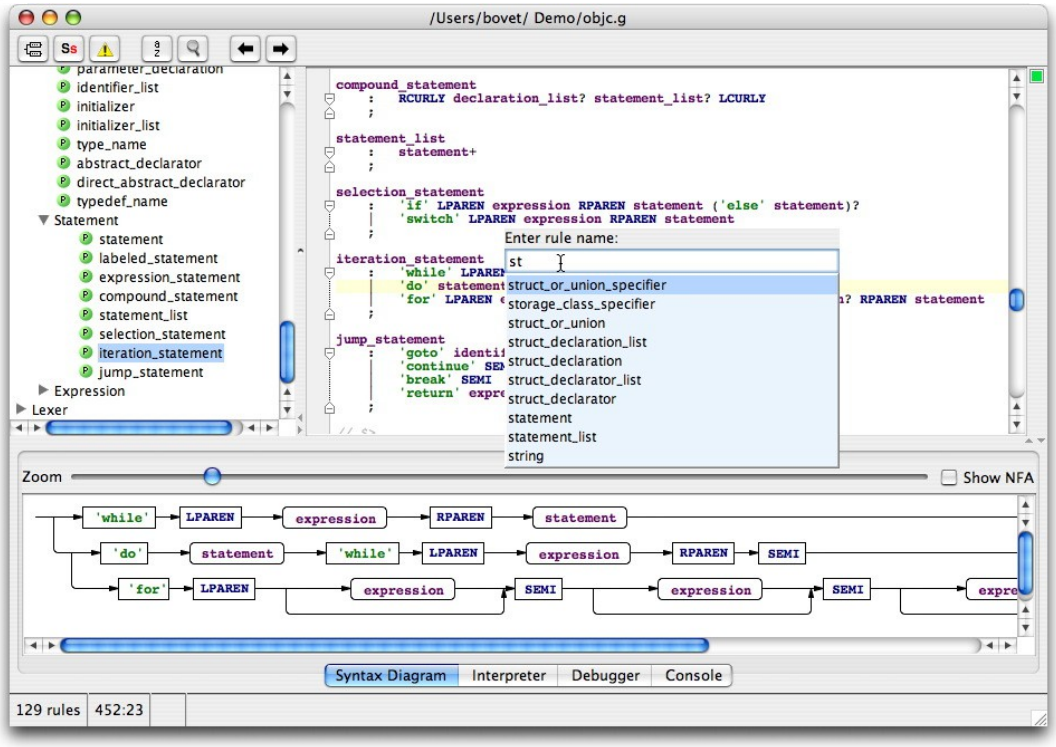
In dem ich aus Grammatiken Parser (Zerteiler und Baumkonstruktoren) generiere

und

zusätzlich Prettyprinter (Codegenerator)

Beispiel EMFText

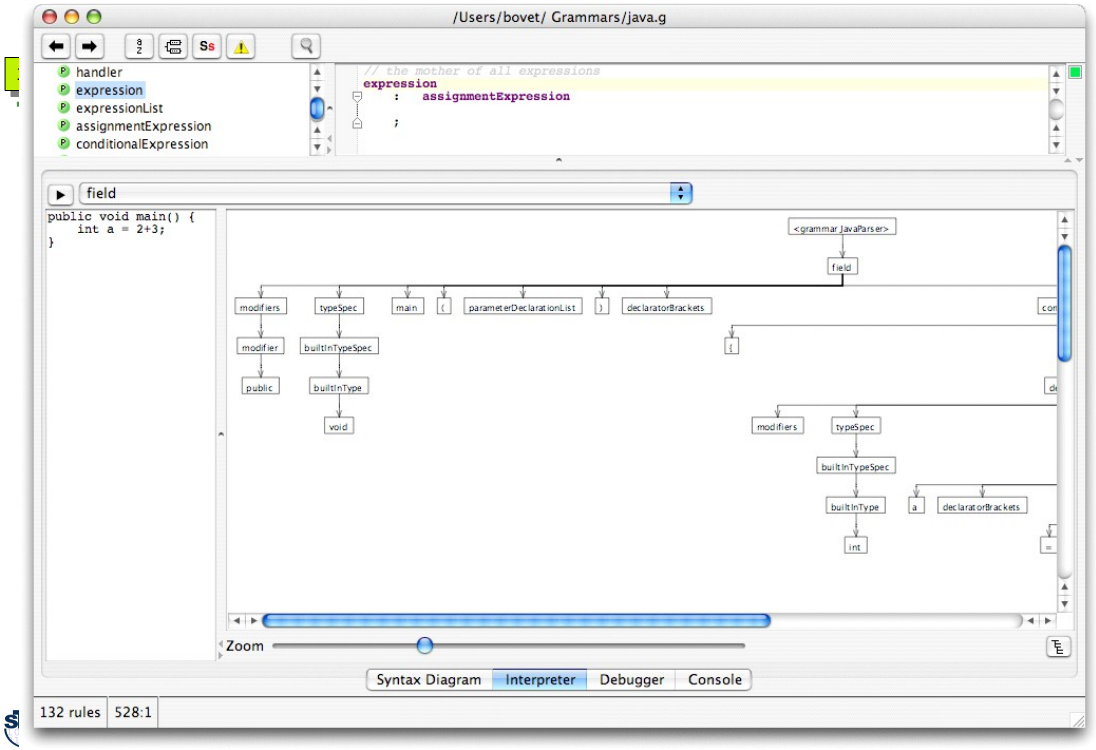
- Nutzt Parser-Generator ANTLR zur Generierung von Parsern
  - Parser und Metamodell werden aufeinander abgebildet (mapping), um konkrete auf abstrakte Syntax abzubilden
- Nutzt schablonengesteuerte Codegenerierung zur Erzeugung von Text und Programmen (siehe später).

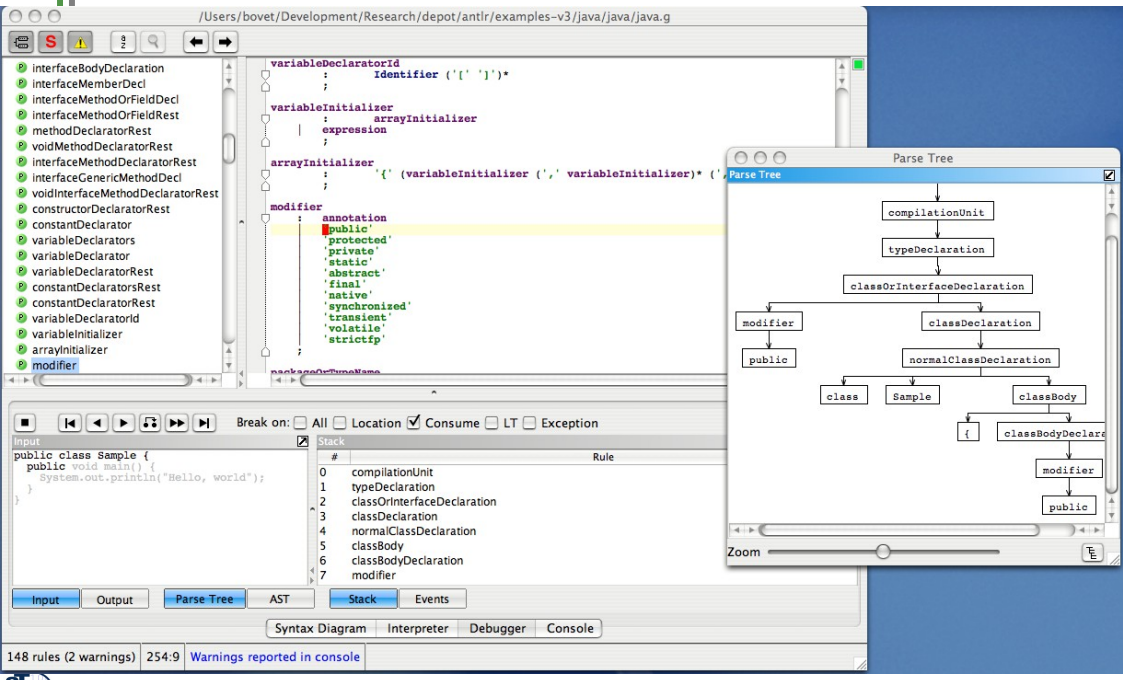


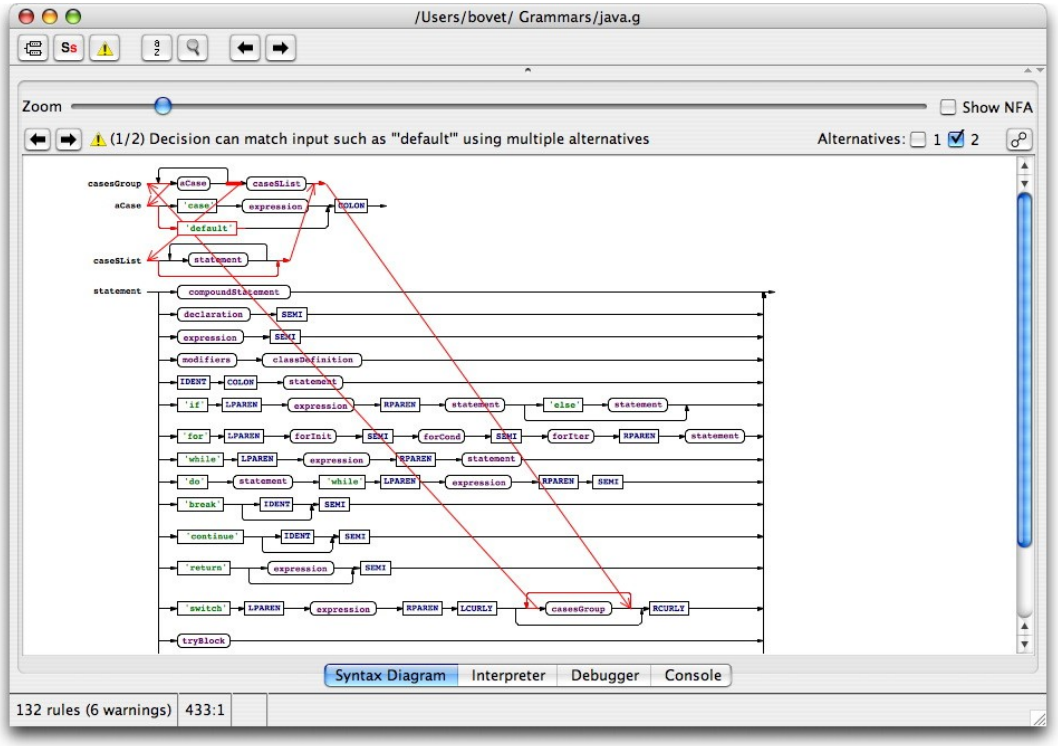
Beispiel: ANTLR www.antlr.org

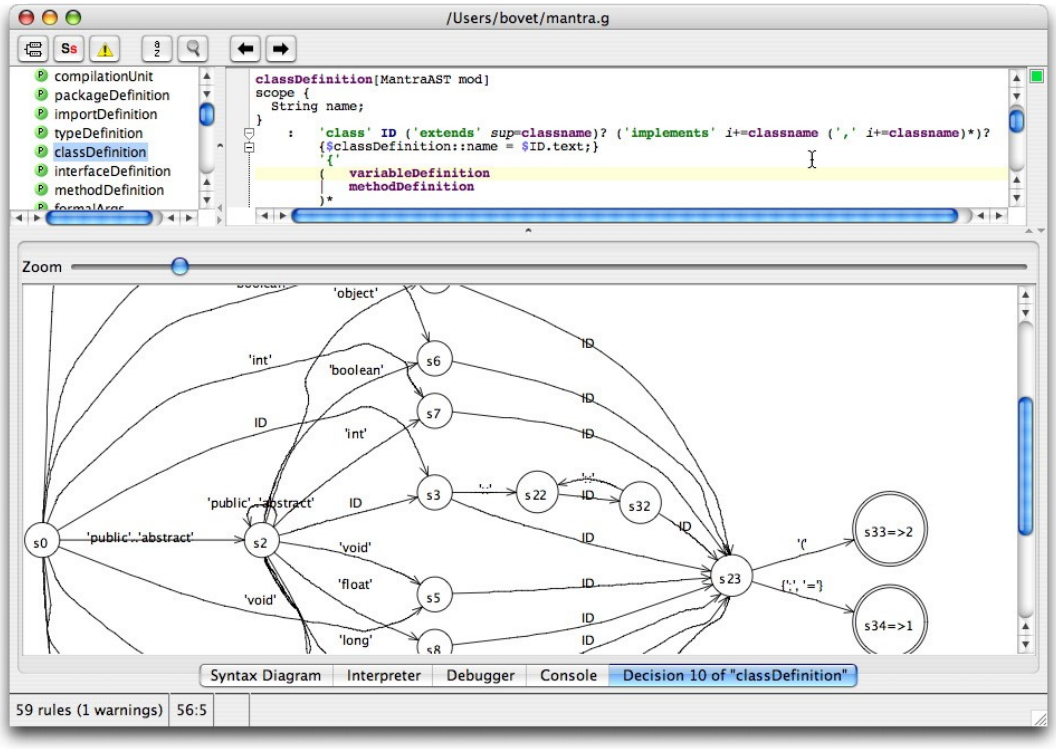
- In den 90ger Jahren gab es für C viele Parsergeneratoren
  - Cocktail's lalr, ell, lark www.cocolab.de
  - fnc2
  - flex und bison (gnu)
- Für Java ist ANTLR populär geworden
  - LL(k)
  - Generierter Parser mit Algorithmus “rekursiver Abstieg”
  - Etwas “gefärbte” Seite mit Geschichte http://www.bearcave.com/software/antlr/antlr_expr.html

Prof. U. Aßmann, Softwareentwicklungswerkzeuge (SEW)







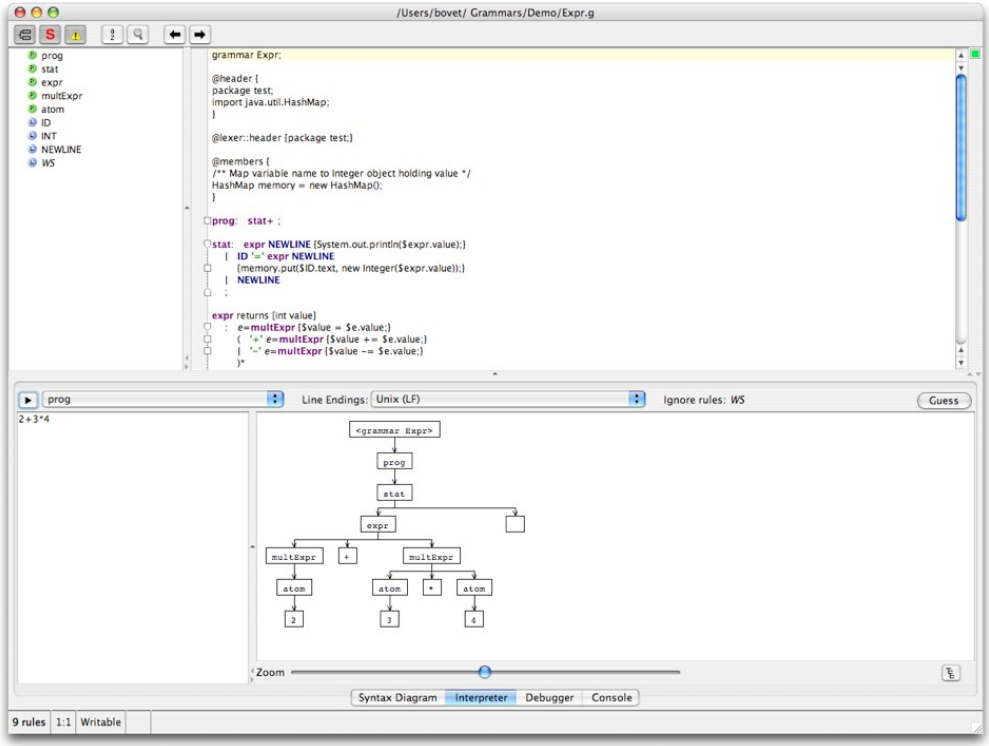


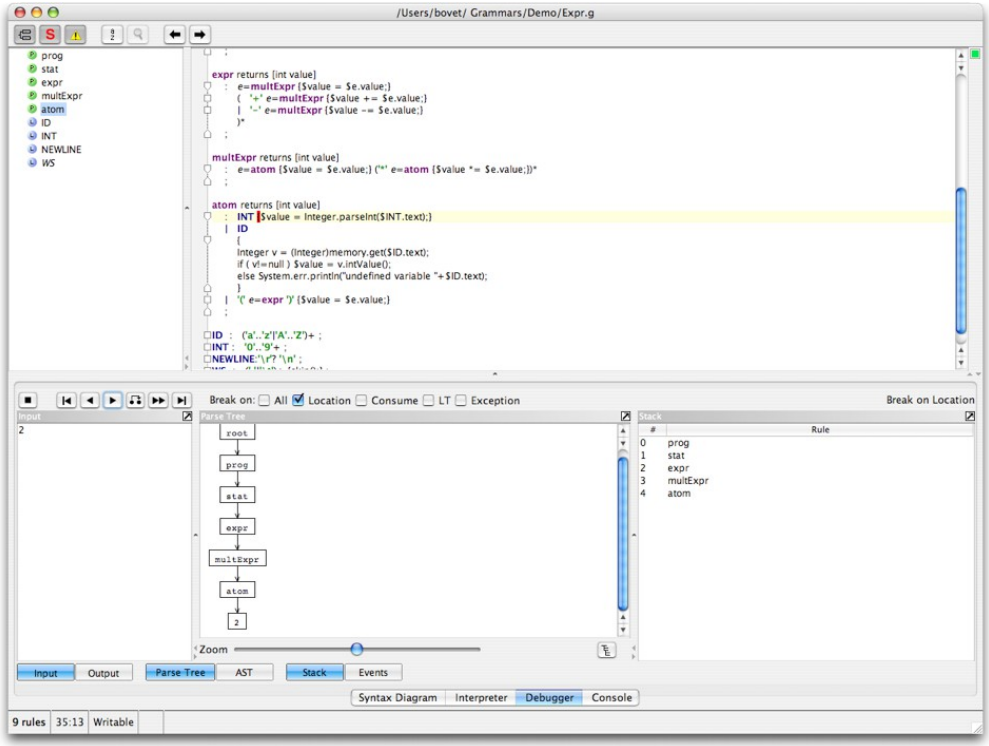
13.2 Ein Taschenrechner

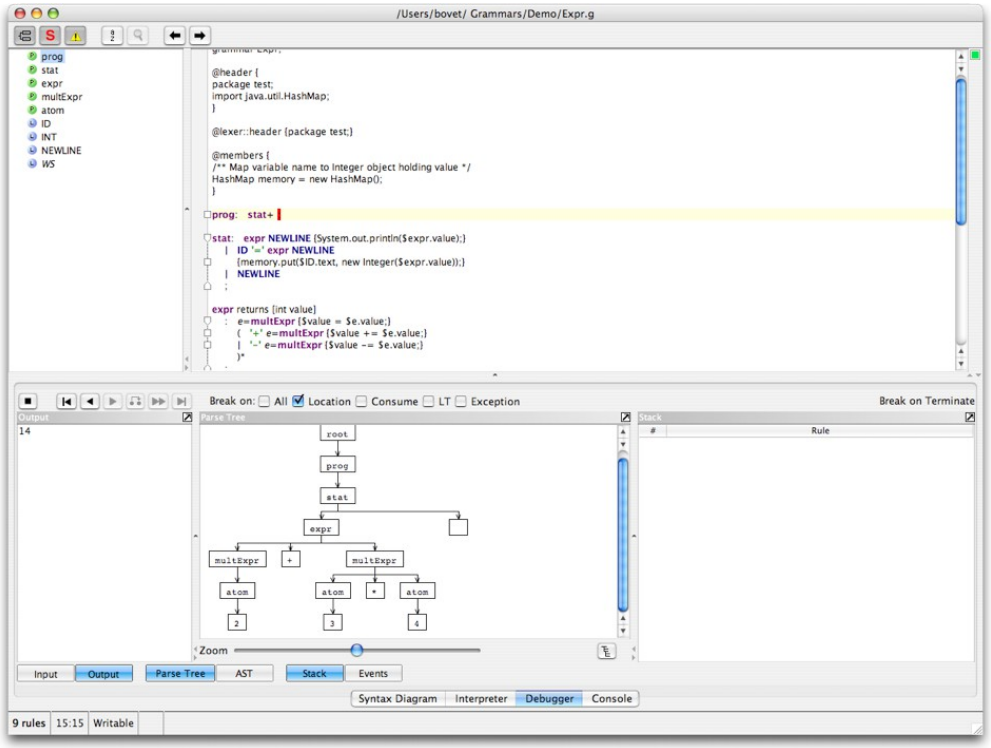
```
grammar Expr;
@header {
package test;
import java.util.HashMap;
}
@lexer::header {package test;}
@members {
/** Map variable name to Integer object holding value */
HashMap memory = new HashMap();
}
prog:   stat+ ;

stat:   expr NEWLINE {System.out.println($expr.value);}
    |   ID '=' expr NEWLINE
        {memory.put($ID.text, new Integer($expr.value));}
    |   NEWLINE
    ;
expr returns [int value]
    :   e=multExpr {$value = $e.value;}
        (   '+' e=multExpr {$value += $e.value;}
        |   '-' e=multExpr {$value -= $e.value;}
        )*
    ;
multExpr returns [int value]
    :   e=atom {$value = $e.value;} ('*' e=atom {$value *= $e.value;})*
    ;
atom returns [int value]
    :   INT {$value = Integer.parseInt($INT.text);}
    |   ID
        {
        Integer v = (Integer)memory.get($ID.text);
        if ( v!=null ) $value = v.intValue();
        else System.err.println("undefined variable "+$ID.text);
        }
    |   '(' e=expr ')' {$value = $e.value;}
    ;
ID  :   ('a'..'z'|'A'..'Z')+ ;
INT :   '0'..'9'+ ;
NEWLINE:'\r'? '\n' ;
WS  :   (' '|'\t')+ {skip();} ;
```


Ansteuerung

```
import org.antlr.runtime.*;
public class Test {
    public static void main(String[] args) throws Exception
{
        ANTLRInputStream input = new
ANTLRInputStream(System.in);
        ExprLexer lexer = new ExprLexer(input);
        CommonTokenStream tokens = new
CommonTokenStream(lexer);
        ExprParser parser = new ExprParser(tokens);
        parser.prog();
    }
}
```





Was haben wir gelernt?

- Parsergeneratoren gehören heute zum Werkzeugsatz jeden Softwareingenieurs
- Neben Cocktail gibt es freie Initiativen, z.B. ANTLR
- Leider erfasst der Parser nur die kontextfeie Struktur des Programms oder Dokuments; Kontextbedingungen und Integritätsbedingungen bleiben der *statischen semantischen Analyse* vorbehalten.

The End