Declare Your Language

Language Definition with the Spoofax Language Workbench
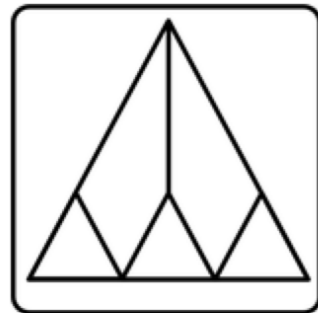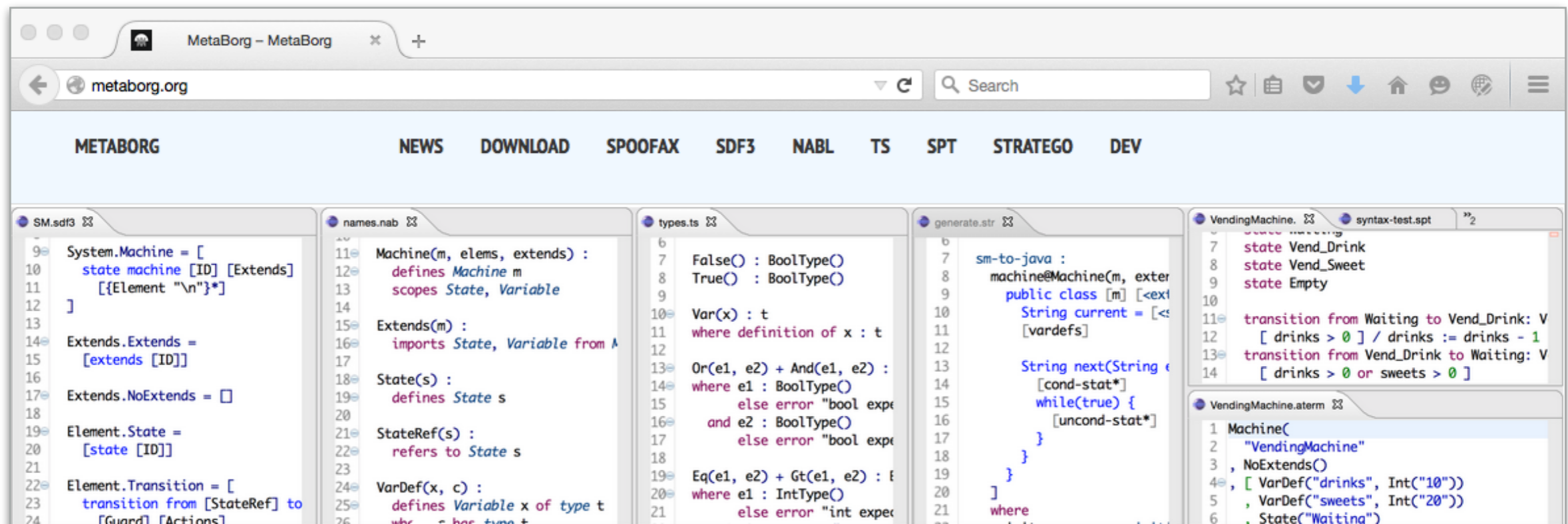
Eelco Visser

DSLDI Summer School, Lausanne
July 13, 2015

TUDelft

Delft University of Technology

SM.sdf3

```
9   System.Machine = [
10      state machine [ID] [Extends]
11      [{Element "\n"}*]
12  ]
13
14  Extends.Extends =
15      [extends [ID]]
16
17  Extends.NoExtends = []
18
19  Element.State =
20      [state [ID]]
21
22  Element.Transition = [
23      transition from [StateRef] to
24      [Guard] [Actions]
```

names.nab

```
11  Machine(m, elems, extends) :
12      defines Machine m
13      scopes State, Variable
14
15  Extends(m) :
16      imports State, Variable from M
17
18  State(s) :
19      defines State s
20
21  StateRef(s) :
22      refers to State s
23
24  VarDef(x, c) :
25      defines Variable x of type t
26      wh     has type t
```

types.ts

```
6
7   False() : BoolType()
8   True()  : BoolType()
9
10  Var(x) : t
11  where definition of x : t
12
13  Or(e1, e2) + And(e1, e2) :
14  where e1 : BoolType()
15      else error "bool exp
16      and e2 : BoolType()
17      else error "bool exp
18
19  Eq(e1, e2) + Gt(e1, e2) : E
20  where e1 : IntType()
21      else error "int expe
```

generate.str

```
7   sm-to-java :
8       machine@Machine(m, exter
9       public class [m] [<ext
10          String current = [<s
11          [vardefs]
12
13          String next(String e
14          [cond-stat*]
15          while(true) {
16              [uncond-stat*]
17          }
18      }
19  }
20  ]
21  where
```

VendingMachine.   syntax-test.spt   »2

```
    state Waiting
7   state Vend_Drink
8   state Vend_Sweet
9   state Empty
10
11  transition from Waiting to Vend_Drink: V
12      [ drinks > 0 ] / drinks := drinks - 1
13  transition from Vend_Drink to Waiting: V
14      [ drinks > 0 or sweets > 0 ]
```

VendingMachine.aterm

```
1  Machine(
2      "VendingMachine"
3  , NoExtends()
4  , [ VarDef("drinks", Int("10"))
5  , VarDef("sweets", Int("20"))
6  , State("Waiting")
```

MetaBorg

MetaBorg is a collection of meta-programming tools including the following components:

- The Spoofax language workbench
- The SDF3 syntax definition formalism
- The NaBL name binding language
- The TS type specification language
- The Stratego transformation language

Software repositories

- https://github.com/metaborg
- https://svn.strategoxt.org/repoman/info/StrategoXT

Edit on GitHub

MetaBorg provides generic technology for allowing a host language (collective) to incorporate and assimilate external domains (cultures) in order to strengthen itself. The ease of implementing embeddings makes resistance futile.

Part I: Hands-On Spoofax Tutorial
- Syntax Definition with SDF3
- Transformation with Stratego
- Name and type analysis with NaBL/TS

Part II: Ongoing Research
- A Theory of Name Resolution
- DynSem: A DSL for Dynamic Semantics Specification