Typify types

Oleg Grenrus

July 23, 2015

Parser and tools for type signatures of typify. Essentially the language is dependent type theory, with omitted lambda constructor.

1 Formal syntax definition

level	name		associativity	example
9	recursive	μ	prefix, right associative	$\mu \ a \to \mu \ b \to a \ b \equiv \mu \ a \to (\mu \ b \to (a \ b))$
8	optional	?	postfix	a?
7	application	a b	left associative	$a b c \equiv (a b) c$
6	conjunction	\wedge	associative	$(a \wedge b) \wedge c \equiv a \wedge (b \wedge c)$
5	disjunction	\vee	associative	$(a \lor b) \lor c \equiv a \lor (b \lor c)$
4	ellipsis		postfix	a
3	type	:	right associative	x:a
2	product	×	associative	$(a \times b) \times c \equiv a \times (b \times c)$
1	function	\rightarrow	right associative	$a \to b \to c \equiv a \to (b \to c)$
0	semicolon	;	associative	, ,

Table 1: Operator precedence

name		code variants
any	T	T *
nothing	\perp	⊥_ _
unit	1	1 ()
optional	?	?
conjunction	\wedge	\wedge
disjunction \(\times \)		V &
ellipsis		(either three dots, or unicode ellipsis)
type	:	:
product	×	× ,
function	\rightarrow	\rightarrow ->
semicolon	;	,
rec	μ	μ rec

Table 2: Special character code representations

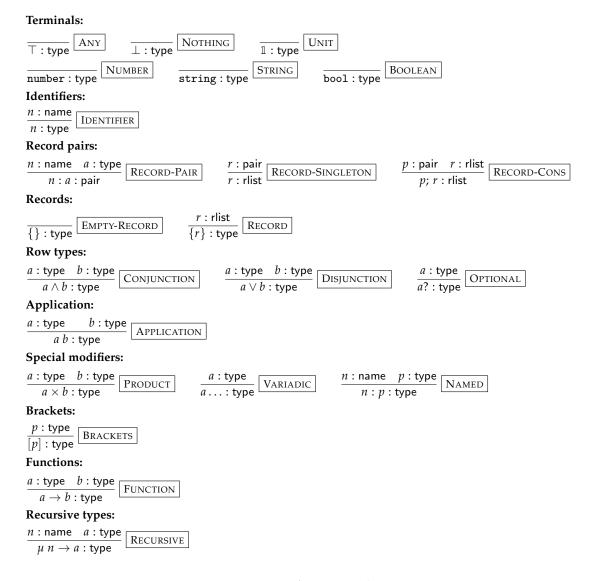


Figure 1: Typify syntax rules

$$a \times ys : b \dots \times c \to d \to e$$

$$\equiv x : a \times ys : (b \dots) \times c \to d \to e$$

$$\equiv (x : a) \times (ys : (b \dots)) \times c \to d \to e$$

$$\equiv ((x : a) \times (ys : (b \dots)) \times c) \to d \to e$$

$$\equiv ((x : a) \times (ys : (b \dots)) \times c) \to d \to e$$

$$\equiv ((x : a) \times (ys : (b \dots)) \times c) \to (d \to e)$$

$$a \times y : b \to c$$

$$\equiv a \times (y : b) \to c$$

$$\equiv (a \times (y : b)) \to c$$

Figure 2: Examples of operator precedence

```
 \textit{Either a b} \equiv \{\textit{type}: \textit{``left''}; \textit{value}: \textit{a}\} \lor \{\textit{type}: \textit{``right''}; \textit{value}: \textit{b}\} \\ \textit{flatMap}: (@:\textit{Observable } \textit{A} \times \textit{f}: \textit{A} \rightarrow \textit{Observable } \textit{B} \lor \textit{Event } \textit{B} \lor \textit{B}) \rightarrow \textit{EventStream } \textit{B}
```

Figure 3: Examples of real world types