An Event-B Specification of

TypingTests

I assume that typing can be made arbitrarily smart, however I do not yet know the limits of how much typing Rodin can do.

For sure both evbt and Rodin does explicit typing based on statements like: ‘x∈N‘ ‘alfa∈N→BOOL‘ or ‘p∈STAFF‘

But Rodin also does implicit typing based on operations. For example:

“‘ @inv1 alfa ∈ ℕ↸BOOL @inv2 beta ∩ ran(alfa) = ∅ “‘

The disjunction forces the type of beta to be the same as the type of ran(alfa) ie ℕ.

“‘ @inv3 x ∈ ℕ @inv4 x+y=7 “‘

The addition forces Rodin the type of y to be ℤ (not ℕ!!)

This projects tests the extent of implicit typing implemented so far in evbt.

1 MACHINE MoreTyping **2**

1.1 *alfa beta x y* 2

1.2 gamma(*b*) 2

MACHINE MoreTyping 5 a 1

VARIABLES 1.1

$alfa$
$beta$
x
y

INVARIANTS

inv1:	$alfa \in \mathbb{N} \nrightarrow \mathrm{BOOL}$	
inv2:	$beta \cap \mathrm{ran}(alfa) = \varnothing$	The type of ran(alfa) will propagate to beta.
inv3:	$x \in \mathbb{N}$	
inv4:	$x + y = 7$	The type of y is deduced to $\mathbb{Z}$.

EVENT INITIALISATION

THEN

init1:	$alfa := \varnothing$
init2:	$beta := \varnothing$
init3:	$x := 14$
init4:	$y := 7$

END

EVENT gamma 1.2

ANY

b

WHERE

grd11:	$b \in beta$

THEN

act11:	$beta := beta \setminus \{b\}$

END

alfa, 2

beta, 2

gamma, 2

INITIALISATION, 2

MoreTyping, 2

x, 2

y, 2