1. Operation: ArgumentTypeModification

Rule1 {

from

Inherit

f : InputMetamodel!InPattern

(

**f.input-reference1**.oclIsKindOf(InputMetaModel!**inputClass**) and

**f.input-reference2**. oclIsKindOf(InPutMetaModel!**inputClass**)

)

to

e : OutputMetamodel!OutPattern (

output-attribute1 <- f.input-attribute1,

output-reference1 <- f.input-reference1

)

}

Since it uses OclISKindOf, right hand side of the OCLisKindOf must inherit from left hand side of it.

Rule2 {

from

n : InputMetamodel!Inpattern

(

**n.input-reference1**->Iteration{exists,select} (edge |

if edge.**input-reference2**.oclIsKindOf(InputMetamodel!**InputClass1**)

)

else

false

**InputClass1 n.input-reference1.input-reference2**

endif

)

)

Inherit

to

activity : OutputMetamodel!OutPattern (

output-reference2 <- right-side)

}

1. Operation: InPatternElementModification

Rule {

from

a : InputMetaModel!InPatternElement

to

d : OutPutMetaModel!OutPatternElement (

output-attribute <- a.input-attribute)

}