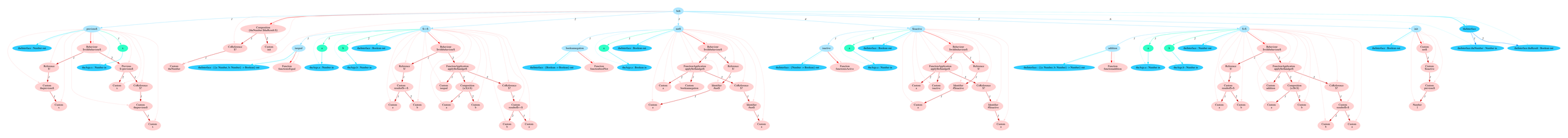

Behaviour
$withbehaviour$

Reference
$!

1

Previous

$=previous$

2

Custom

theprevious$

1

Custom

x

1

CoReference
$?

1

Custom

x

2

Custom

theprevious$

1

Custom

x

1

Function
functionisEqual

Behaviour
$withbehaviour$

Reference
$!

1

FunctionApplication
applytoandget$

2

Custom
resultof$==$

1

Custom

a

1

Custom
b

2

Custom

isequal

1

Composition
{a:$,b:$}

2

CoReference
$?

3

Custom

a

1

Custom
b

2

Custom
resultof$==$

1

Custom

a

1

Custom
b

2

Function
functionboolNot

Behaviour
$withbehaviour$

Reference
$!

1

FunctionApplication
applytoandget$

2

Identifier
#not$

1

Custom

a

1

Custom

booleannegation

1

Custom

a

2

CoReference
$?

3

Identifier
#not$

1

Custom

a

1

Function
functionisActive

Behaviour
$withbehaviour$

Reference
$!

1

FunctionApplication
applytoandget$

2

Identifier
#$isactive

1

Custom

a

1

Custom
isactive

1

Custom

a

2

CoReference
$?

3

Identifier
#$isactive

1

Custom

a

1

Function
functionaddition

Behaviour
$withbehaviour$

Reference
$!

1

FunctionApplication
applytoandget$

2

Custom
resultof$+$

1

Custom

a

1

Custom
b

2

Custom
addition

1

Composition
{a:$b:$}

2

CoReference
$?

3

Custom

a

1

Custom
b

2

Custom
resultof$+$

1

Custom

a

1

Custom
b

2

Custom
not$

Custom
$isactive

1

Custom

previous$

1

Number
1

1

Composition
{theNumber:$theResult:$}

CoReference
$?

1

Custom
init

2

Custom
theNumber

1

bob

previous$

1

$==$

2

not$

3

$isactive

4

$+$

5

init

6

theInterface

x theInterface.theNumber : Number in theInterface.theResult : Boolean out

1

theArgs.x : Number in

theInterface : Number out isequal

1

a

1

b

2

theArgs.a : Number in theArgs.b : Number in

theInterface : Boolean out

theInterface : {{a: Number, b: Number} -> Boolean} out

booleannegation

1

a

1

theArgs.a : Boolean in

theInterface : Boolean out

theInterface : {Boolean -> Boolean} out

isactive

1

a

1

theArgs.a : Number in

theInterface : Boolean out

theInterface : {Number -> Boolean} out

addition

1

a

1

b

2

theArgs.a : Number in theArgs.b : Number in

theInterface : Number out

theInterface : {{a: Number, b: Number} -> Number} out

theInterface : Boolean out

1 2