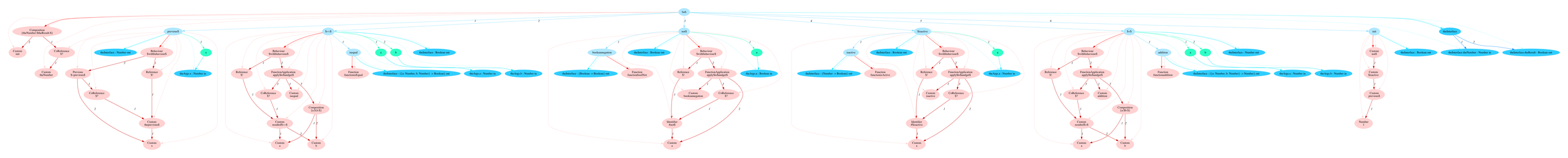

Custom

x

Custom

theprevious$

1

Reference
$!

1 CoReference

$?

1

Previous

$=previous$

2

1

Behaviour
$withbehaviour$

2 1

Function
functionisEqual

Custom

a

Custom
b

Custom
resultof$==$

1 2

Reference
$!

1 Custom
isequal

Composition
{a:$,b:$}

1 2

CoReference
$?

1

FunctionApplication
applytoandget$

3 1 2

Behaviour
$withbehaviour$

1 2

Function
functionboolNot

Custom

a

Identifier
#not$

1

Reference
$!

1 Custom
booleannegation

CoReference
$?

1

FunctionApplication
applytoandget$

2

1 3

Behaviour
$withbehaviour$

1 2

Function
functionisActive

Custom

a

Identifier
#$isactive

1

Reference
$!

1 Custom

isactive

CoReference
$?

1

FunctionApplication
applytoandget$

2

1 3

Behaviour
$withbehaviour$

1 2

Function
functionaddition

Custom

a

Custom
b

Custom
resultof$+$

1 2

Reference
$!

1 Custom

addition

Composition
{a:$b:$}

1 2

CoReference
$?

1

FunctionApplication
applytoandget$

3 1 2

Behaviour
$withbehaviour$

1 2

Number
1

Custom

previous$

1

Custom
$isactive

1

Custom
not$

1

Custom
theNumber

CoReference
$?

1

Custom
init

Composition
{theNumber:$theResult:$}

2 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