

Basic Classes – API Changes

(Status 22-10-2014)

The recent introduction of new tools for developing and simulating POOSL models included changes in the API of the basic classes. The new version (version 3.0) is described in detail in the POOSL language reference available from <http://poosl.esi.nl/documentation/>

This document describes the changes between version 2.0 and version 3.0 of the basic classes:

Version 2.0	Version 3.0	Remarks
Object	Object	Extended
Array	Array	Changed & Extended
String	String	Changed & Extended
Nil	Nil	No Changes
Boolean	Boolean	Extended
Char	Char	Extended
Integer	Integer	Changed & Extended
Real	Real	Changed & Extended
RandomGenerator	RandomGenerator	Extended
FileIn	FileIn	Changed & Extended
FileOut	FileOut	Changed & Extended
Socket (SHESim only)	Socket	Changed, Extended & Fully Supported
	Observer	New & Not Yet Fully Supported
	Console	New & Not Yet Fully Supported
Number		Deprecated
Display (SHESim only)		Deprecated
Console (SHESim only)		Deprecated

The next sections give an overview of the changes to the methods for each of the basic classes.

Object

Version 2.0	Version 3.0	Remarks
!=	!=	No Changes
!==(!==(No Changes
=	=	No Changes

==	==	No Changes
deepCopy	deepCopy	No Changes
error	error	No Changes
	assert	New
	marchal	New / Not Yet Fully Supported
printString	printString	No Changes
shallowCopy	shallowCopy	No Changes
	isOfType	New

Array

An important difference between version 2.0 and version 3.0 is that *Array*'s are not automatically resized anymore when putting an object at an index larger than the size of the *Array*. Notice also the changed meaning of method *size*.

Version 2.0	Version 3.0	Remarks
printString	printString	Difference in resulting String
get	at	Renamed
getSize	size	Renamed
put	putAt	Renamed & No Automatic Resizing
putAll	putAll	No Changes
size	resize	Renamed
	+	New
	concat	New
	find	New
	subArray	New

String

An important extension in version 3.0 is support for escape sequences as common in programming languages.

Version 2.0	Version 3.0	Remarks
+ append	+	Returns new String
concat	concat	Modifies receiver
cr	cr	No Changes

	lf	New
tab	tab	No Changes
	find	New
get	at	Renamed
length	size	Renamed
put	putAt	Renamed
copy	subString	Renamed
	unmarchal	New / Not Yet Fully Supported
	splitOn	New
	splitOnWhiteSpace	New
	isBoolean	New
	isChar	New
	isReal	New
	isInteger	New
	toBoolean	New
	toChar	New
	toReal	New
	toInteger	New
<	<	No Changes
<=	<=	No Changes
>	>	No Changes
>=	>=	No Changes

Boolean

The syntax now also support expressions where prefix symbol ! is used as Boolean negation.

Version 2.0	Version 3.0	Remarks
&	&	No Changes
		No Changes
not	not !	No Changes & Extra Syntax
	xor	New

Char

An important extension in version 3.0 is support for escape sequences as common in programming languages.

Version 2.0	Version 3.0	Remarks
	asciiIndex	New
	asString	New

Integer

An important difference between version 2.0 and version 3.0 is that operations on *Integers* are limited to *Integer* parameters only (instead of also *Reals*). *Reals* can be used as parameters to operators on *Integers* by using method `asInteger`.

Version 2.0	Version 3.0	Remarks
- (negation)	- (negation)	No Changes
- (subtraction)	- (subtraction)	Integer Parameter Only
*	*	Integer Parameter Only
&	&	No Changes
+	+	Integer Parameter Only
<	<	Integer Parameter Only
<=	<=	Integer Parameter Only
>	>	Integer Parameter Only
>=	>=	Integer Parameter Only
		No Changes
abs	abs	No Changes
	asAsciiChar	New
asReal	asReal	No Changes
div	div	No Changes
/	/	Integer Parameter Only
fac	fac	No Changes
modulo	modulo	Integer Parameter Only
monus	monus	No Changes
not	not	No Changes
power	power	Integer Parameter Only
sqr	sqr	No Changes
	max	New

	min	New
	xor	New
sqrt		Deprecated
asInteger		Deprecated
asString		Deprecated (Use printString)

Real

An important difference between Version 2.0 and Version 3.0 is that operations on *Reals* are limited to *Real* parameters only (instead of also *Integers*). *Integers* can be used as parameters for operators on *Reals* by using method `asReal`.

Version 2.0	Version 3.0	Remarks
- (negation)	- (negation)	No Changes
- (subtraction)	- (subtraction)	Real Parameter Only
*	*	Real Parameter Only
/	/	Real Parameter Only
+	+	Real Parameter Only
<	<	Real Parameter Only
<=	<=	Real Parameter Only
>	>	Real Parameter Only
>=	>=	Real Parameter Only
abs	abs	No Changes
acos	acos	No Changes
asin	asin	No Changes
asInteger	asInteger	No Changes
atan	atan	No Changes
atan2	atan2	No Changes
ceiling	ceiling	No Changes
cos	cos	No Changes
exp	exp	No Changes
floor	floor	No Changes
ln	ln	No Changes
log	log	No Changes
	monus	New
power	power	Real Parameter Only

round	round	No Changes
sin	sin	No Changes
sqr	sqr	No Changes
sqrt	sqrt	No Changes
tan	tan	No Changes
	max	New
	min	New
div		Deprecated
modulo		Deprecated
asReal		Deprecated
asString		Deprecated (Use printString)

RandomGenerator

Version 2.0	Version 3.0	Remarks
random	random	No Changes
randomInt	randomInt	No Changes
randomiseSeed	randomiseSeed	No Changes
setSeed	seed	Renamed

FileIn

An important change between version 2.0 and version 3.0 is that Strings read from file do not need to be between double quotes “ anymore (except for the new version of method readString which reads a syntactic representation of a String variable, which is between double quotes “, see the POOSL language reference available from <http://poosl.esi.nl/documentation/>).

Version 2.0	Version 3.0	Remarks
source	source	No Changes
open	open	No Changes
	atEndOfFile	New
close	close	No Changes
	read	New
	readUntil	New
	readWord	New
	readLine	New
readString	readString	Changed to Syntactic

	Representation of Strings
readInteger	Deprecated (Use for example readWord with String::isInteger and String::toInteger)
readNumber	Deprecated (Use for example readWord with String::isInteger and String::toInteger or String::isReal and String::toReal)
readReal	Deprecated (Use for example readWord with String::isReal and String::toReal)
readBoolean	Deprecated (Use for example readWord with String::isBoolean and String::toBoolean)
readChar	Deprecated (Use for example readWord with String::isChar and String::toChar)

FileOut

Notice especially the changed meaning of method *writeString*.

Version 2.0	Version 3.0	Remarks
destination	destination	No Changes
open	open	No Changes
append	append	No Changes
	flush	New
close	close	No Changes
writeString	write	Renamed
writeln	writeln	Takes String as Parameter (Use writeln(O printString) in Version 3.0 to replace writeln(O) from Version 2.0)
	writeString	Takes String as Parameter (Writes its syntactic representation to the file)
Write		Deprecated (Use write(O printString tab) in Version 3.0 to replace write(O) from Version 2.0)

Socket

Version 2.0	Version 3.0	Remarks
connectFrom	acceptFrom	Renamed
connectTo	connectTo	No Changes
isConnected	isConnected	No Changes
isDisconnected	isDisconnected	No Changes
disconnect	close	Renamed
hasInput	hasCharacters	Renamed & Single Parameter
read	read	No Changes
	hasCharacter	New
	readUntil	New
hasInputWord	hasWord	Renamed
readWord	readWord	No Changes
	hasLine	New
	readLine	New
hasInputString	hasString	Renamed
readString	readString	No Changes
write	write	No Changes
	writeLine	New
	writeString	New
isConnecting		Deprecated
isDisconnecting		Deprecated