Laboratory for Advanced Software Systems
University of Luxembourg

excalibur Standard Libraries Documentation
- v 1.4 -

(*Report type: Specification*)

Tuesday 2nd February, 2016 - 13:37

Contents

List of Figures

Listings

Chapter 1

Introduction

Chapter 2

General Description

2.1 Use Cases Model

This section contains the use cases elicited during the requirements elicitation phase. The use cases are textually described as suggested by the Messir method and inspired by the standard Cokburn template [?].

2.1.1 Use Cases

There are no elements in this category in the system analysed.

2.1.2 Use Case Instance(s)

There are no elements in this category in the system analysed.

Chapter 3

Environment Model

3.1 Environment model view(s)

There are no view(s) for the Messir environment model.

3.2 Actors and Interfaces Descriptions

There are no elements in this category in the system analysed.

Chapter 4

Concept Model

4.1 PrimaryTypes-Datatypes

4.1.1 Local view 01

Figure 4.1 math-local-view-01

4.1.2 Local view 02

Figure 4.2 calendar-local-view-01

4.1.3 Local view 03

Figure 4.3 string-local-view-01

4.1.4 Local view 04

Figure 4.4 primitives-local-view-01

4.2 Concept Model Types Descriptions

This section provides the textual descriptions of all the types defined in the concept model and that can be part of the graphical views provided.

4.2.1 Primary types - Class types descriptions

There are no elements in this category in the system analysed.

4.2.2 Primary types - Datatypes types descriptions

There are no elements in this category in the system analysed.

4.2.3 Primary types - Association types descriptions

There are no association types for the primary types.

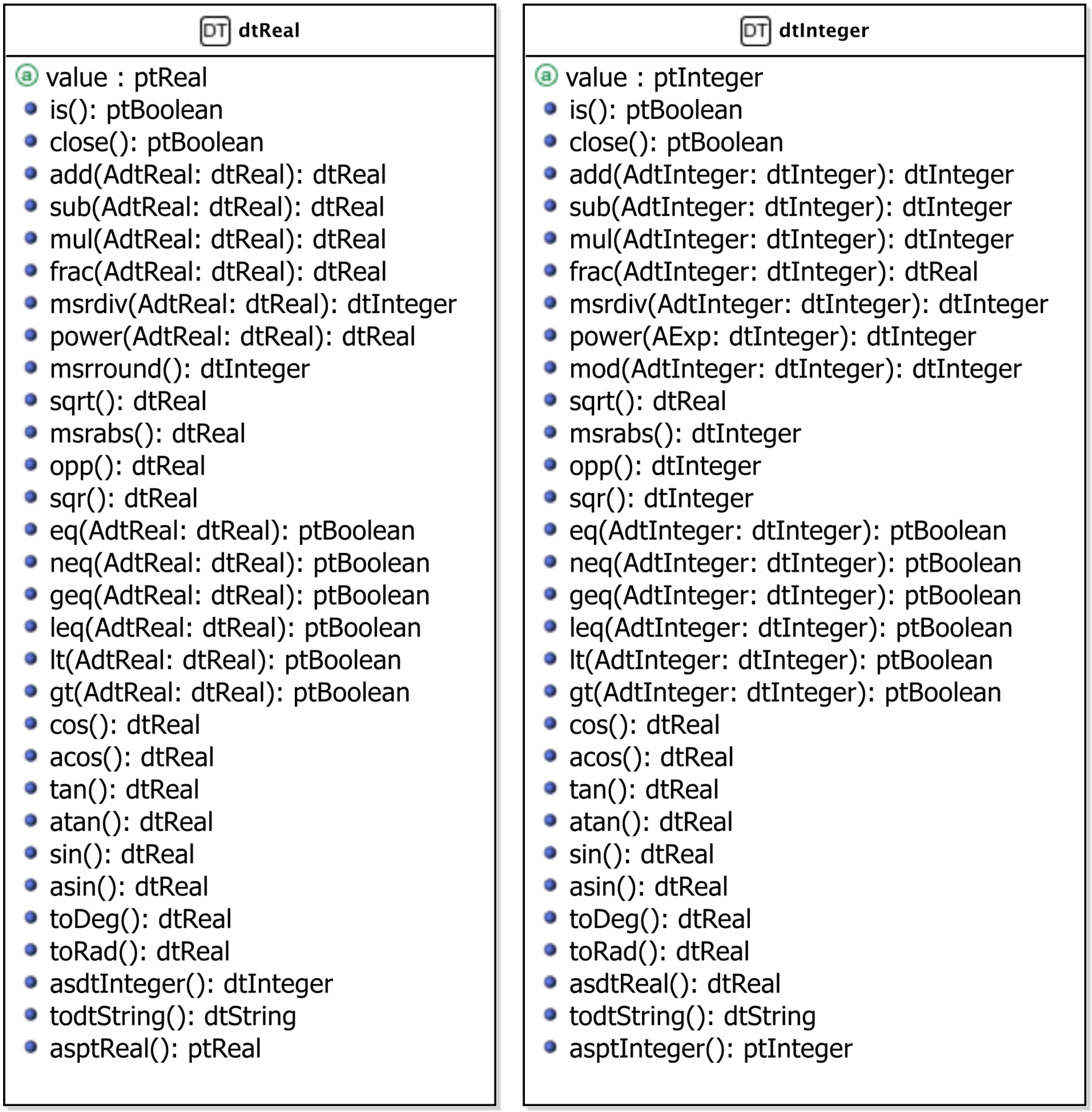
DT dtReal
value : ptReal
is(): ptBoolean
close(): ptBoolean
add(AdtReal: dtReal): dtReal
sub(AdtReal: dtReal): dtReal
mul(AdtReal: dtReal): dtReal
frac(AdtReal: dtReal): dtReal
msrdiv(AdtReal: dtReal): dtInteger
power(AdtReal: dtReal): dtReal
msrround(): dtInteger
sqrt(): dtReal
msrabs(): dtReal
opp(): dtReal
sqr(): dtReal
eq(AdtReal: dtReal): ptBoolean
neq(AdtReal: dtReal): ptBoolean
geq(AdtReal: dtReal): ptBoolean
leq(AdtReal: dtReal): ptBoolean
lt(AdtReal: dtReal): ptBoolean
gt(AdtReal: dtReal): ptBoolean
cos(): dtReal
acos(): dtReal
tan(): dtReal
atan(): dtReal
sin(): dtReal
asin(): dtReal
toDeg(): dtReal
toRad(): dtReal
asdtInteger(): dtInteger
todtString(): dtString
asptReal(): ptReal
DT dtInteger
value : ptInteger
is(): ptBoolean
close(): ptBoolean
add(AdtInteger: dtInteger): dtInteger
sub(AdtInteger: dtInteger): dtInteger
mul(AdtInteger: dtInteger): dtInteger
frac(AdtInteger: dtInteger): dtReal
msrdiv(AdtInteger: dtInteger): dtInteger
power(AExp: dtInteger): dtInteger
mod(AdtInteger: dtInteger): dtInteger
sqrt(): dtReal
msrabs(): dtInteger
opp(): dtInteger
sqr(): dtInteger
eq(AdtInteger: dtInteger): ptBoolean
neq(AdtInteger: dtInteger): ptBoolean
geq(AdtInteger: dtInteger): ptBoolean
leq(AdtInteger: dtInteger): ptBoolean
lt(AdtInteger: dtInteger): ptBoolean
gt(AdtInteger: dtInteger): ptBoolean
cos(): dtReal
acos(): dtReal
tan(): dtReal
atan(): dtReal
sin(): dtReal
asin(): dtReal
toDeg(): dtReal
toRad(): dtReal
asdtReal(): dtReal
todtString(): dtString
asptInteger(): ptInteger

Figure 4.1: Concept Model - PrimaryTypes-Datatypes local view 01. .

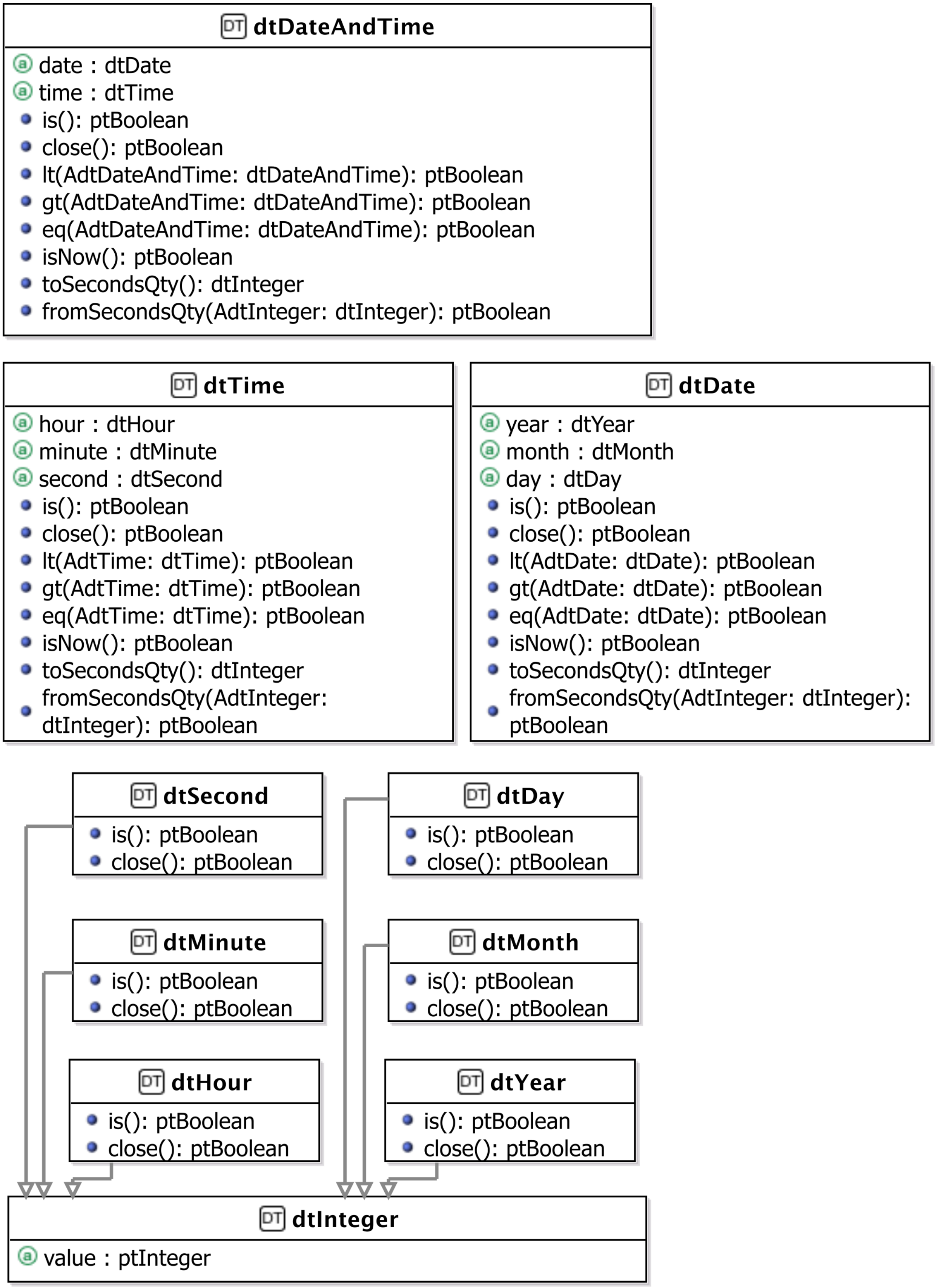

Figure 4.2: Concept Model - PrimaryTypes-Datatypes local view 02. .

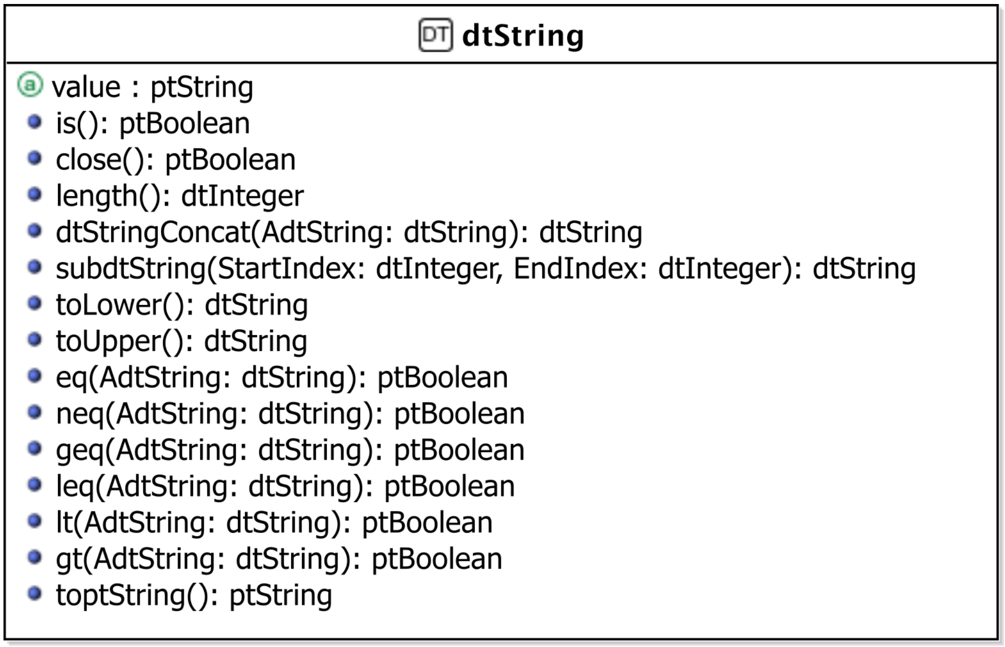
DT dtString
value : ptString
is(): ptBoolean
close(): ptBoolean
length(): dtInteger
dtStringConcat(AdtString: dtString): dtString
subdtString(StartIndex: dtInteger, EndIndex: dtInteger): dtString
toLower(): dtString
toUpper(): dtString
eq(AdtString: dtString): ptBoolean
neq(AdtString: dtString): ptBoolean
geq(AdtString: dtString): ptBoolean
leq(AdtString: dtString): ptBoolean
lt(AdtString: dtString): ptBoolean
gt(AdtString: dtString): ptBoolean
toptString(): ptString

Figure 4.3: Concept Model - PrimaryTypes-Datatypes local view 03. .

4.2.4 Primary types - Aggregation types descriptions

There are no aggregation types for the primary types.

4.2.4.1 Primary types - Composition types descriptions

There are no composition types for the primary types.

4.2.5 Secondary types - Class types descriptions

There are no elements in this category in the system analysed.

4.2.6 Secondary types - Datatypes types descriptions

There are no elements in this category in the system analysed.

4.2.7 Secondary types - Association types descriptions

There are no association types for the secondary types.

4.2.8 Secondary types - Aggregation types descriptions

There are no aggregation types for the secondary types.

4.2.9 Secondary types - Composition types descriptions

There are no composition types for the secondary types.

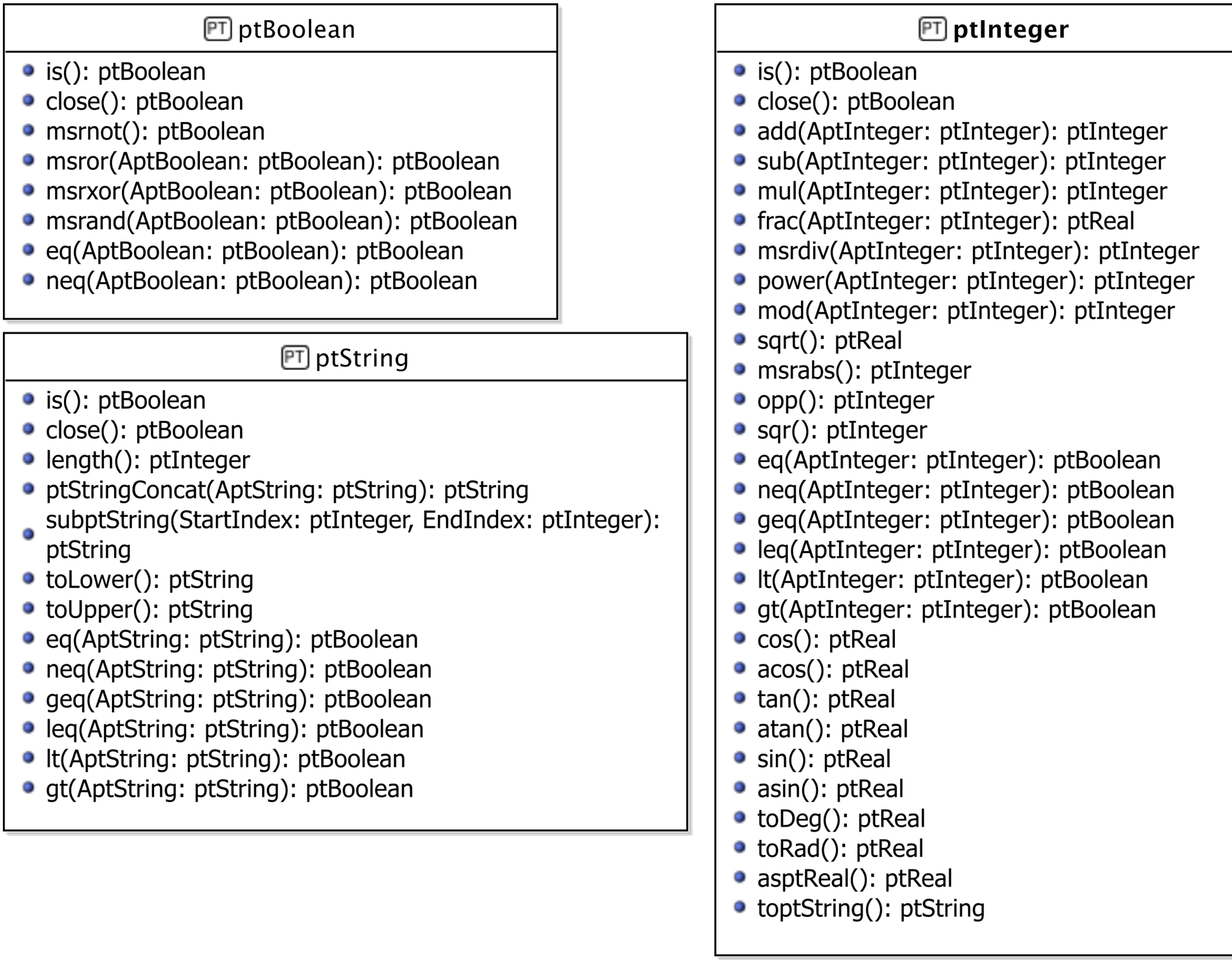
PT ptBoolean
is(): ptBoolean
close(): ptBoolean
msrnot(): ptBoolean
msror(AptBoolean: ptBoolean): ptBoolean
msrxor(AptBoolean: ptBoolean): ptBoolean
msrand(AptBoolean: ptBoolean): ptBoolean
eq(AptBoolean: ptBoolean): ptBoolean
neq(AptBoolean: ptBoolean): ptBoolean
PT ptString
is(): ptBoolean
close(): ptBoolean
length(): ptInteger
ptStringConcat(AptString: ptString): ptString
subptString(StartIndex: ptInteger, EndIndex: ptInteger): ptString
toLower(): ptString
toUpper(): ptString
eq(AptString: ptString): ptBoolean
neq(AptString: ptString): ptBoolean
geq(AptString: ptString): ptBoolean
leq(AptString: ptString): ptBoolean
lt(AptString: ptString): ptBoolean
gt(AptString: ptString): ptBoolean
PT ptInteger
is(): ptBoolean
close(): ptBoolean
add(AptInteger: ptInteger): ptInteger
sub(AptInteger: ptInteger): ptInteger
mul(AptInteger: ptInteger): ptInteger
frac(AptInteger: ptInteger): ptReal
msrdiv(AptInteger: ptInteger): ptInteger
power(AptInteger: ptInteger): ptInteger
mod(AptInteger: ptInteger): ptInteger
sqrt(): ptReal
msrabs(): ptInteger
opp(): ptInteger
sqr(): ptInteger
eq(AptInteger: ptInteger): ptBoolean
neq(AptInteger: ptInteger): ptBoolean
geq(AptInteger: ptInteger): ptBoolean
leq(AptInteger: ptInteger): ptBoolean
lt(AptInteger: ptInteger): ptBoolean
gt(AptInteger: ptInteger): ptBoolean
cos(): ptReal
acos(): ptReal
tan(): ptReal
atan(): ptReal
sin(): ptReal
asin(): ptReal
toDeg(): ptReal
toRad(): ptReal
asptReal(): ptReal
toptString(): ptString

Figure 4.4: Concept Model - PrimaryTypes-Datatypes local view 04. .

Chapter 5

Operation Model

This section contains the operation schemes of each operation defined in either an actor, its output interface, in a primary or secondary type (class, datatype or enumeration types). The Messir OCL code listing is joined to the comment table.

5.1 Environment - Out Interface Operation Schemes

There are no elements in this category in the system analysed.

5.2 Environment - Actor Operation Schemes

There are no elements in this category in the system analysed.

5.3 Primary Types - Operation Schemes for Classes

There are no elements in this category in the system analysed.

5.4 Primary Types - Operation Schemes for Datatypes

There are no elements in this category in the system analysed.

5.5 Primary Types - Operation Schemes for Enumerations

There are no elements in this category in the system analysed.

5.6 Secondary Types - Operation Schemes for Classes

There are no elements in this category in the system analysed.

5.7 Secondary Types - Operation Schemes for Datatypes

There are no elements in this category in the system analysed.

5.8 Secondary Types - Operation Schemes for Enumerations

There are no elements in this category in the system analysed.

Chapter 6

Test Model(s)

There are no elements in this category in the system analysed.

Chapter 7

Additional Constraints

Appendix A

Undocumented Messir Specification Elements

A.1 Undocumented Primary Types

A.1.1 Undocumented Primary Datatype Types

- lu.uni.lassy.messir.libraries.calendar.dtDate
- lu.uni.lassy.messir.libraries.calendar.dtDateAndTime
- lu.uni.lassy.messir.libraries.calendar.dtDay
- lu.uni.lassy.messir.libraries.calendar.dtHour
- lu.uni.lassy.messir.libraries.math.dtInteger
- lu.uni.lassy.messir.libraries.calendar.dtMinute
- lu.uni.lassy.messir.libraries.calendar.dtMonth
- lu.uni.lassy.messir.libraries.math.dtReal
- lu.uni.lassy.messir.libraries.calendar.dtSecond
- lu.uni.lassy.messir.libraries.string.dtString
- lu.uni.lassy.messir.libraries.calendar.dtTime
- lu.uni.lassy.messir.libraries.calendar.dtYear

A.1.2 Undocumented Primary Primitive Types

- lu.uni.lassy.messir.libraries.primitives.ptBoolean
- lu.uni.lassy.messir.libraries.primitives.ptInteger
- lu.uni.lassy.messir.libraries.primitives.ptReal
- lu.uni.lassy.messir.libraries.primitives.ptString

A.2 Undocumented Operation Specifications

- lu.uni.lassy.messir.libraries.calendar.dtDate.close
- lu.uni.lassy.messir.libraries.calendar.dtDate.eq
- lu.uni.lassy.messir.libraries.calendar.dtDate.fromSecondsQty
- lu.uni.lassy.messir.libraries.calendar.dtDate.gt
- lu.uni.lassy.messir.libraries.calendar.dtDate.is
- lu.uni.lassy.messir.libraries.calendar.dtDate.isNow
- lu.uni.lassy.messir.libraries.calendar.dtDate.lt
- lu.uni.lassy.messir.libraries.calendar.dtDate.toSecondsQty
- lu.uni.lassy.messir.libraries.calendar.dtDateAndTime.close
- lu.uni.lassy.messir.libraries.calendar.dtDateAndTime.eq
- lu.uni.lassy.messir.libraries.calendar.dtDateAndTime.fromSecondsQty
- lu.uni.lassy.messir.libraries.calendar.dtDateAndTime.gt
- lu.uni.lassy.messir.libraries.calendar.dtDateAndTime.is
- lu.uni.lassy.messir.libraries.calendar.dtDateAndTime.isNow
- lu.uni.lassy.messir.libraries.calendar.dtDateAndTime.lt
- lu.uni.lassy.messir.libraries.calendar.dtDateAndTime.toSecondsQty
- lu.uni.lassy.messir.libraries.calendar.dtDay.close
- lu.uni.lassy.messir.libraries.calendar.dtDay.is
- lu.uni.lassy.messir.libraries.calendar.dtHour.close
- lu.uni.lassy.messir.libraries.calendar.dtHour.is
- lu.uni.lassy.messir.libraries.math.dtInteger.acos
- lu.uni.lassy.messir.libraries.math.dtInteger.add
- lu.uni.lassy.messir.libraries.math.dtInteger.asdtReal
- lu.uni.lassy.messir.libraries.math.dtInteger.asin
- lu.uni.lassy.messir.libraries.math.dtInteger.asptInteger
- lu.uni.lassy.messir.libraries.math.dtInteger.atan
- lu.uni.lassy.messir.libraries.math.dtInteger.close
- lu.uni.lassy.messir.libraries.math.dtInteger.cos
- lu.uni.lassy.messir.libraries.math.dtInteger.eq

- lu.uni.lassy.messir.libraries.math.dtInteger.frac
- lu.uni.lassy.messir.libraries.math.dtInteger.geq
- lu.uni.lassy.messir.libraries.math.dtInteger.gt
- lu.uni.lassy.messir.libraries.math.dtInteger.is
- lu.uni.lassy.messir.libraries.math.dtInteger.leq
- lu.uni.lassy.messir.libraries.math.dtInteger.lt
- lu.uni.lassy.messir.libraries.math.dtInteger.mod
- lu.uni.lassy.messir.libraries.math.dtInteger.msrabs
- lu.uni.lassy.messir.libraries.math.dtInteger.msrdiv
- lu.uni.lassy.messir.libraries.math.dtInteger.mul
- lu.uni.lassy.messir.libraries.math.dtInteger.neq
- lu.uni.lassy.messir.libraries.math.dtInteger.opp
- lu.uni.lassy.messir.libraries.math.dtInteger.power
- lu.uni.lassy.messir.libraries.math.dtInteger.sin
- lu.uni.lassy.messir.libraries.math.dtInteger.sqr
- lu.uni.lassy.messir.libraries.math.dtInteger.sqrt
- lu.uni.lassy.messir.libraries.math.dtInteger.sub
- lu.uni.lassy.messir.libraries.math.dtInteger.tan
- lu.uni.lassy.messir.libraries.math.dtInteger.toDeg
- lu.uni.lassy.messir.libraries.math.dtInteger.toRad
- lu.uni.lassy.messir.libraries.math.dtInteger.todtString
- lu.uni.lassy.messir.libraries.calendar.dtMinute.close
- lu.uni.lassy.messir.libraries.calendar.dtMinute.is
- lu.uni.lassy.messir.libraries.calendar.dtMonth.close
- lu.uni.lassy.messir.libraries.calendar.dtMonth.is
- lu.uni.lassy.messir.libraries.math.dtReal.acos
- lu.uni.lassy.messir.libraries.math.dtReal.add
- lu.uni.lassy.messir.libraries.math.dtReal.asdtInteger
- lu.uni.lassy.messir.libraries.math.dtReal.asin
- lu.uni.lassy.messir.libraries.math.dtReal.asptReal

- lu.uni.lassy.messir.libraries.math.dtReal.atan
- lu.uni.lassy.messir.libraries.math.dtReal.close
- lu.uni.lassy.messir.libraries.math.dtReal.cos
- lu.uni.lassy.messir.libraries.math.dtReal.eq
- lu.uni.lassy.messir.libraries.math.dtReal.frac
- lu.uni.lassy.messir.libraries.math.dtReal.geq
- lu.uni.lassy.messir.libraries.math.dtReal.gt
- lu.uni.lassy.messir.libraries.math.dtReal.is
- lu.uni.lassy.messir.libraries.math.dtReal.leq
- lu.uni.lassy.messir.libraries.math.dtReal.lt
- lu.uni.lassy.messir.libraries.math.dtReal.msrabs
- lu.uni.lassy.messir.libraries.math.dtReal.msrdiv
- lu.uni.lassy.messir.libraries.math.dtReal.msrround
- lu.uni.lassy.messir.libraries.math.dtReal.mul
- lu.uni.lassy.messir.libraries.math.dtReal.neq
- lu.uni.lassy.messir.libraries.math.dtReal.opp
- lu.uni.lassy.messir.libraries.math.dtReal.power
- lu.uni.lassy.messir.libraries.math.dtReal.sin
- lu.uni.lassy.messir.libraries.math.dtReal.sqr
- lu.uni.lassy.messir.libraries.math.dtReal.sqrt
- lu.uni.lassy.messir.libraries.math.dtReal.sub
- lu.uni.lassy.messir.libraries.math.dtReal.tan
- lu.uni.lassy.messir.libraries.math.dtReal.toDeg
- lu.uni.lassy.messir.libraries.math.dtReal.toRad
- lu.uni.lassy.messir.libraries.math.dtReal.todtString
- lu.uni.lassy.messir.libraries.calendar.dtSecond.close
- lu.uni.lassy.messir.libraries.calendar.dtSecond.is
- lu.uni.lassy.messir.libraries.string.dtString.close
- lu.uni.lassy.messir.libraries.string.dtString.dtStringConcat
- lu.uni.lassy.messir.libraries.string.dtString.eq

- lu.uni.lassy.messir.libraries.string.dtString.geq
- lu.uni.lassy.messir.libraries.string.dtString.gt
- lu.uni.lassy.messir.libraries.string.dtString.is
- lu.uni.lassy.messir.libraries.string.dtString.length
- lu.uni.lassy.messir.libraries.string.dtString.leq
- lu.uni.lassy.messir.libraries.string.dtString.lt
- lu.uni.lassy.messir.libraries.string.dtString.neq
- lu.uni.lassy.messir.libraries.string.dtString.subdtString
- lu.uni.lassy.messir.libraries.string.dtString.toLower
- lu.uni.lassy.messir.libraries.string.dtString.toUpper
- lu.uni.lassy.messir.libraries.string.dtString.toptString
- lu.uni.lassy.messir.libraries.calendar.dtTime.close
- lu.uni.lassy.messir.libraries.calendar.dtTime.eq
- lu.uni.lassy.messir.libraries.calendar.dtTime.fromSecondsQty
- lu.uni.lassy.messir.libraries.calendar.dtTime.gt
- lu.uni.lassy.messir.libraries.calendar.dtTime.is
- lu.uni.lassy.messir.libraries.calendar.dtTime.isNow
- lu.uni.lassy.messir.libraries.calendar.dtTime.lt
- lu.uni.lassy.messir.libraries.calendar.dtTime.toSecondsQty
- lu.uni.lassy.messir.libraries.calendar.dtYear.close
- lu.uni.lassy.messir.libraries.calendar.dtYear.is
- lu.uni.lassy.messir.libraries.primitives.ptBoolean.close
- lu.uni.lassy.messir.libraries.primitives.ptBoolean.eq
- lu.uni.lassy.messir.libraries.primitives.ptBoolean.is
- lu.uni.lassy.messir.libraries.primitives.ptBoolean.msrand
- lu.uni.lassy.messir.libraries.primitives.ptBoolean.msrnot
- lu.uni.lassy.messir.libraries.primitives.ptBoolean.msror
- lu.uni.lassy.messir.libraries.primitives.ptBoolean.msrxor
- lu.uni.lassy.messir.libraries.primitives.ptBoolean.neq
- lu.uni.lassy.messir.libraries.primitives.ptInteger.acos

- lu.uni.lassy.messir.libraries.primitives.ptInteger.add
- lu.uni.lassy.messir.libraries.primitives.ptInteger.asin
- lu.uni.lassy.messir.libraries.primitives.ptInteger.asptReal
- lu.uni.lassy.messir.libraries.primitives.ptInteger.atan
- lu.uni.lassy.messir.libraries.primitives.ptInteger.close
- lu.uni.lassy.messir.libraries.primitives.ptInteger.cos
- lu.uni.lassy.messir.libraries.primitives.ptInteger.eq
- lu.uni.lassy.messir.libraries.primitives.ptInteger.frac
- lu.uni.lassy.messir.libraries.primitives.ptInteger.geq
- lu.uni.lassy.messir.libraries.primitives.ptInteger.gt
- lu.uni.lassy.messir.libraries.primitives.ptInteger.is
- lu.uni.lassy.messir.libraries.primitives.ptInteger.leq
- lu.uni.lassy.messir.libraries.primitives.ptInteger.lt
- lu.uni.lassy.messir.libraries.primitives.ptInteger.mod
- lu.uni.lassy.messir.libraries.primitives.ptInteger.msrabs
- lu.uni.lassy.messir.libraries.primitives.ptInteger.msrdiv
- lu.uni.lassy.messir.libraries.primitives.ptInteger.mul
- lu.uni.lassy.messir.libraries.primitives.ptInteger.neq
- lu.uni.lassy.messir.libraries.primitives.ptInteger.opp
- lu.uni.lassy.messir.libraries.primitives.ptInteger.power
- lu.uni.lassy.messir.libraries.primitives.ptInteger.sin
- lu.uni.lassy.messir.libraries.primitives.ptInteger.sqr
- lu.uni.lassy.messir.libraries.primitives.ptInteger.sqrt
- lu.uni.lassy.messir.libraries.primitives.ptInteger.sub
- lu.uni.lassy.messir.libraries.primitives.ptInteger.tan
- lu.uni.lassy.messir.libraries.primitives.ptInteger.toDeg
- lu.uni.lassy.messir.libraries.primitives.ptInteger.toRad
- lu.uni.lassy.messir.libraries.primitives.ptInteger.toptString
- lu.uni.lassy.messir.libraries.primitives.ptReal.acos
- lu.uni.lassy.messir.libraries.primitives.ptReal.add

- lu.uni.lassy.messir.libraries.primitives.ptReal.asin
- lu.uni.lassy.messir.libraries.primitives.ptReal.asptInteger
- lu.uni.lassy.messir.libraries.primitives.ptReal.atan
- lu.uni.lassy.messir.libraries.primitives.ptReal.close
- lu.uni.lassy.messir.libraries.primitives.ptReal.cos
- lu.uni.lassy.messir.libraries.primitives.ptReal.eq
- lu.uni.lassy.messir.libraries.primitives.ptReal.frac
- lu.uni.lassy.messir.libraries.primitives.ptReal.geq
- lu.uni.lassy.messir.libraries.primitives.ptReal.gt
- lu.uni.lassy.messir.libraries.primitives.ptReal.is
- lu.uni.lassy.messir.libraries.primitives.ptReal.leq
- lu.uni.lassy.messir.libraries.primitives.ptReal.lt
- lu.uni.lassy.messir.libraries.primitives.ptReal.msrabs
- lu.uni.lassy.messir.libraries.primitives.ptReal.msrdiv
- lu.uni.lassy.messir.libraries.primitives.ptReal.msrround
- lu.uni.lassy.messir.libraries.primitives.ptReal.mul
- lu.uni.lassy.messir.libraries.primitives.ptReal.neq
- lu.uni.lassy.messir.libraries.primitives.ptReal.opp
- lu.uni.lassy.messir.libraries.primitives.ptReal.power
- lu.uni.lassy.messir.libraries.primitives.ptReal.sin
- lu.uni.lassy.messir.libraries.primitives.ptReal.sqr
- lu.uni.lassy.messir.libraries.primitives.ptReal.sqrt
- lu.uni.lassy.messir.libraries.primitives.ptReal.sub
- lu.uni.lassy.messir.libraries.primitives.ptReal.tan
- lu.uni.lassy.messir.libraries.primitives.ptReal.toDeg
- lu.uni.lassy.messir.libraries.primitives.ptReal.toRad
- lu.uni.lassy.messir.libraries.primitives.ptReal.toptString
- lu.uni.lassy.messir.libraries.primitives.ptString.close
- lu.uni.lassy.messir.libraries.primitives.ptString.eq
- lu.uni.lassy.messir.libraries.primitives.ptString.geq

- lu.uni.lassy.messir.libraries.primitives.ptString.gt
- lu.uni.lassy.messir.libraries.primitives.ptString.is
- lu.uni.lassy.messir.libraries.primitives.ptString.length
- lu.uni.lassy.messir.libraries.primitives.ptString.leq
- lu.uni.lassy.messir.libraries.primitives.ptString.lt
- lu.uni.lassy.messir.libraries.primitives.ptString.neq
- lu.uni.lassy.messir.libraries.primitives.ptString.ptStringConcat
- lu.uni.lassy.messir.libraries.primitives.ptString.subptString
- lu.uni.lassy.messir.libraries.primitives.ptString.toLower
- lu.uni.lassy.messir.libraries.primitives.ptString.toUpper

Appendix B

Messir Specification Files Listing

B.1 File ./src-gen/messir-spec/.views.msr

```
//
//DON'T TOUCH THIS FILE !!!
//
package uuid7d4b15133efc45b9b0f503fbb2d93068 {
 Concept Model {}
}
```

Listing B.1: Messir Spec. file .views.msr.

B.2 File ./src-gen/messir-spec/library/calendar.msr

```
/*
* Copyright University of Luxembourg
*
* This file is part of EXCALIBUR.
* EXCALIBUR is free software: you can redistribute it and/or modify
* it under the terms of the GNU General Public License as published by
* the Free Software Foundation, version 3 of the License.
*
* EXCALIBUR is distributed in the hope that it will be useful,
* but WITHOUT ANY WARRANTY; without even the implied warranty of
* MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
* GNU General Public License for more details.
*
* You should have received a copy of the GNU General Public License
* along with EXCALIBUR in the COPYING.txt file.
* If not, see <http://www.gnu.org/licenses/>.
*
* Last Modification:
*
* @author nicolas.guelfi
* @date Mon May 06 18:10:54 CEST 2013
*/

package lu.uni.lassy.messir.libraries.calendar{

import lu.uni.lassy.messir.libraries.primitives
import lu.uni.lassy.messir.libraries.math

 Concept Model {

  Primary Types {

   datatype dtHour extends dtInteger {
    operation is():ptBoolean
    external operation close() : ptBoolean
   }
```

```
datatype dtMinute extends dtInteger {
 operation is():ptBoolean
 external operation close() : ptBoolean
}
datatype dtSecond extends dtInteger {
 operation is():ptBoolean
 external operation close() : ptBoolean
}

datatype dtTime {
 attribute hour:dtHour
 attribute minute: dtMinute
 attribute second: dtSecond

 operation is():ptBoolean
 external operation close() : ptBoolean

 // Logical Operations
 operation lt(AdtTime:dtTime):ptBoolean
 operation gt(AdtTime:dtTime):ptBoolean
 operation eq(AdtTime:dtTime):ptBoolean
 external operation isNow():ptBoolean

 // Conversion Operations
 operation toSecondsQty():dtInteger
 operation fromSecondsQty(AdtInteger:dtInteger):ptBoolean
}

datatype dtYear extends dtInteger {
 operation is():ptBoolean
 external operation close() : ptBoolean
}
datatype dtMonth extends dtInteger {
 operation is():ptBoolean
 external operation close() : ptBoolean
}
datatype dtDay extends dtInteger {
 operation is():ptBoolean
 external operation close() : ptBoolean
}

datatype dtDate {
 attribute year:dtYear
 attribute month: dtMonth
 attribute day: dtDay

 operation is():ptBoolean
 external operation close() : ptBoolean

 // Logical Operations
 operation lt(AdtDate:dtDate):ptBoolean
 operation gt(AdtDate:dtDate):ptBoolean
 operation eq(AdtDate:dtDate):ptBoolean
 external operation isNow():ptBoolean

 // Conversion Operations
 operation toSecondsQty():dtInteger
 operation fromSecondsQty(AdtInteger:dtInteger):ptBoolean

}
datatype dtDateAndTime {
 attribute date:dtDate
 attribute time: dtTime

 operation is():ptBoolean
 external operation close() : ptBoolean

 operation lt(AdtDateAndTime:dtDateAndTime):ptBoolean
 operation gt(AdtDateAndTime:dtDateAndTime):ptBoolean
 operation eq(AdtDateAndTime:dtDateAndTime):ptBoolean
```

```
    external operation isNow():ptBoolean

    // Conversion Operations
    operation toSecondsQty():dtInteger
    operation fromSecondsQty(AdtInteger:dtInteger):ptBoolean
   }
  }
 }
}
```

Listing B.2: Messir Spec. file calendar.msr.

B.3 File ./src-gen/messir-spec/library/collections.msr

```
package lu.uni.lassy.messir.libraries.collections {

import lu.uni.lassy.messir.libraries.primitives

 Concept Model {

  Primary Types {
    generictype col(T){
    external operation colSize() : ptInteger
    external operation colIsEmpty() : ptBoolean
    external operation colSum() : ptInteger
    external operation colAny() : T
    external operation colSort() : col(T)
   }
   }
   }
}
```

Listing B.3: Messir Spec. file collections.msr.

B.4 File ./src-gen/messir-spec/library/math.msr

```
/*
* Copyright University of Luxembourg
*
* This file is part of EXCALIBUR.
* EXCALIBUR is free software: you can redistribute it and/or modify
* it under the terms of the GNU General Public License as published by
* the Free Software Foundation, version 3 of the License.
*
* EXCALIBUR is distributed in the hope that it will be useful,
* but WITHOUT ANY WARRANTY; without even the implied warranty of
* MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
* GNU General Public License for more details.
*
* You should have received a copy of the GNU General Public License
* along with EXCALIBUR in the COPYING.txt file.
* If not, see <http://www.gnu.org/licenses/>.
*
* Last Modification:
*
* @author nicolas.guelfi
* @date Mon May 06 18:10:54 CEST 2013
*/

package lu.uni.lassy.messir.libraries.math{

import lu.uni.lassy.messir.libraries.primitives
import lu.uni.lassy.messir.libraries.string

 Concept Model {

  Primary Types {

```

```
datatype dtInteger{
 attribute value:ptInteger
 // Type checking Operation
 operation is():ptBoolean
 external operation close() : ptBoolean

 // Arithmetic Operations
 operation add(AdtInteger:dtInteger): dtInteger
 operation sub(AdtInteger:dtInteger): dtInteger
 operation mul(AdtInteger:dtInteger): dtInteger
 operation frac(AdtInteger:dtInteger): dtReal
 operation msrdiv(AdtInteger:dtInteger): dtInteger
 operation power(AExp:dtInteger): dtInteger
 operation mod(AdtInteger:dtInteger): dtInteger

 operation sqrt(): dtReal
 operation msrabs(): dtInteger
 operation opp(): dtInteger
 operation sqr(): dtInteger

 // Logical Operations
 operation eq(AdtInteger:dtInteger): ptBoolean
 operation neq(AdtInteger:dtInteger): ptBoolean
 operation geq(AdtInteger:dtInteger): ptBoolean
 operation leq(AdtInteger:dtInteger): ptBoolean
 operation lt(AdtInteger:dtInteger): ptBoolean
 operation gt(AdtInteger:dtInteger): ptBoolean

 // Trigonometric Operations
 // default is radian
 operation cos(): dtReal
 operation acos(): dtReal
 operation tan(): dtReal
 operation atan(): dtReal
 operation sin(): dtReal
 operation asin(): dtReal
 operation toDeg(): dtReal
 operation toRad(): dtReal

 // Conversion Operations
 operation asdtReal():dtReal
 operation todtString():dtString
 operation asptInteger():ptInteger
 }

datatype dtReal {
 attribute value:ptReal

 // Type checking Operation
 operation is():ptBoolean
 external operation close() : ptBoolean

 // Arithmetic Operations
 operation add(AdtReal:dtReal): dtReal
 operation sub(AdtReal:dtReal): dtReal
 operation mul(AdtReal:dtReal): dtReal
 operation frac(AdtReal:dtReal) : dtReal
 operation msrdiv(AdtReal:dtReal): dtInteger
 operation power(AdtReal:dtReal): dtReal

 operation msrround() : dtInteger
 operation sqrt(): dtReal
 operation msrabs(): dtReal
 operation opp(): dtReal
 operation sqr(): dtReal

 // Logical Operations
 operation eq(AdtReal:dtReal): ptBoolean
 operation neq(AdtReal:dtReal): ptBoolean
 operation geq(AdtReal:dtReal): ptBoolean
```

```
    operation leq(AdtReal:dtReal): ptBoolean
    operation lt(AdtReal:dtReal): ptBoolean
    operation gt(AdtReal:dtReal): ptBoolean

    // Trigonometric Operations
    // default is radian
    operation cos(): dtReal
    operation acos(): dtReal
    operation tan(): dtReal
    operation atan(): dtReal
    operation sin(): dtReal
    operation asin(): dtReal
    operation toDeg(): dtReal
    operation toRad(): dtReal

    // Conversion Operations
    operation asdtInteger():dtInteger
     operation todtString() : dtString
    operation asptReal():ptReal
   }
  }
 }
}
```

Listing B.4: Messir Spec. file math.msr.

B.5 File ./src-gen/messir-spec/library/primitives.msr

```
/*
* Copyright University of Luxembourg
*
* This file is part of EXCALIBUR.
* EXCALIBUR is free software: you can redistribute it and/or modify
* it under the terms of the GNU General Public License as published by
* the Free Software Foundation, version 3 of the License.
*
* EXCALIBUR is distributed in the hope that it will be useful,
* but WITHOUT ANY WARRANTY; without even the implied warranty of
* MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
* GNU General Public License for more details.
*
* You should have received a copy of the GNU General Public License
* along with EXCALIBUR in the COPYING.txt file.
* If not, see <http://www.gnu.org/licenses/>.
*
* Last Modification:
*
* @author nicolas.guelfi
* @date Mon May 06 18:10:54 CEST 2013
*/

package lu.uni.lassy.messir.libraries.primitives{
 Concept Model {
  Primary Types {

 primitive ptBoolean {
   external operation is() : ptBoolean
   external operation close() : ptBoolean

   external operation msrnot() : ptBoolean
   external operation msror(AptBoolean:ptBoolean) : ptBoolean
   external operation msrxor(AptBoolean:ptBoolean) : ptBoolean
   external operation msrand(AptBoolean:ptBoolean) : ptBoolean
   external operation eq(AptBoolean:ptBoolean) : ptBoolean
   external operation neq(AptBoolean:ptBoolean) : ptBoolean
   }

 primitive ptInteger {

  operation is() : ptBoolean
  external operation close() : ptBoolean

   // Arithmetic Operations
  external operation add(AptInteger:ptInteger) : ptInteger
  external operation sub(AptInteger:ptInteger) : ptInteger
  external operation mul(AptInteger:ptInteger) : ptInteger
  external operation frac(AptInteger:ptInteger) : ptReal
  external operation msrdiv(AptInteger:ptInteger) : ptInteger
  external operation power(AptInteger:ptInteger) : ptInteger
  external operation mod(AptInteger:ptInteger) : ptInteger

  external operation sqrt() : ptReal
  external operation msrabs() : ptInteger
  external operation opp() : ptInteger
  external operation sqr() : ptInteger

   // Logical Operations
  external operation eq(AptInteger:ptInteger) : ptBoolean
  external operation neq(AptInteger:ptInteger) : ptBoolean
  external operation geq(AptInteger:ptInteger) : ptBoolean
  external operation leq(AptInteger:ptInteger) : ptBoolean
  external operation lt(AptInteger:ptInteger) : ptBoolean
  external operation gt(AptInteger:ptInteger) : ptBoolean

  // Trigonometric Operations
  // default is radian
  external operation cos(): ptReal
  external operation acos(): ptReal
  external operation tan(): ptReal
  external operation atan(): ptReal
  external operation sin(): ptReal
  external operation asin(): ptReal
  external operation toDeg(): ptReal
  external operation toRad(): ptReal

   // Conversion Operations
  external operation asptReal() : ptReal
  external operation toptString() : ptString
}

primitive ptReal {

  operation is() : ptBoolean
  external operation close() : ptBoolean

  // Arithmetic Operations
  external operation add(AptReal:ptReal) : ptReal
  external operation sub(AptReal:ptReal) : ptReal
  external operation mul(AptReal:ptReal) : ptReal
  external operation frac(AptReal:ptReal) : ptReal
  external operation msrdiv(AptReal:ptReal) : ptInteger
  external operation power(AptReal:ptReal) : ptReal

  external operation msrround() : ptInteger
  external operation sqrt() : ptReal
  external operation msrabs() : ptReal
  external operation opp() : ptReal
  external operation sqr() : ptReal

   // Logical Operations
  external operation eq(AptReal:ptReal) : ptBoolean
  external operation neq(AptReal:ptReal) : ptBoolean
  external operation geq(AptReal:ptReal) : ptBoolean
  external operation leq(AptReal:ptReal) : ptBoolean
  external operation lt(AptReal:ptReal) : ptBoolean
  external operation gt(AptReal:ptReal) : ptBoolean

   // Trigonometric Operations
   // default is radian
   external operation cos(): ptReal
   external operation acos(): ptReal
   external operation tan(): ptReal
   external operation atan(): ptReal
   external operation sin(): ptReal
   external operation asin(): ptReal
   external operation toDeg(): ptReal
   external operation toRad(): ptReal

    // Conversion Operations
   external operation asptInteger() : ptInteger
   external operation toptString() : ptString
}

primitive ptString {

   external operation is() : ptBoolean
   external operation close() : ptBoolean

   external operation length() : ptInteger
   external operation ptStringConcat(AptString:ptString) : ptString
   external operation subptString(
             StartIndex:ptInteger,
             EndIndex:ptInteger
             ) : ptString
   external operation toLower():ptString
   external operation toUpper():ptString
   external operation eq(AptString:ptString):ptBoolean
   external operation neq(AptString:ptString):ptBoolean
   external operation geq(AptString:ptString) : ptBoolean
   external operation leq(AptString:ptString) : ptBoolean
   external operation lt(AptString:ptString) : ptBoolean
   external operation gt(AptString:ptString) : ptBoolean
   }
 }
}
}
```

Listing B.5: Messir Spec. file primitives.msr.

B.6 File ./src-gen/messir-spec/library/string.msr

```
/*
* Copyright University of Luxembourg
*
* This file is part of EXCALIBUR.
* EXCALIBUR is free software: you can redistribute it and/or modify
* it under the terms of the GNU General Public License as published by
* the Free Software Foundation, version 3 of the License.
*
* EXCALIBUR is distributed in the hope that it will be useful,
* but WITHOUT ANY WARRANTY; without even the implied warranty of
* MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
* GNU General Public License for more details.
*
* You should have received a copy of the GNU General Public License
* along with EXCALIBUR in the COPYING.txt file.
* If not, see <http://www.gnu.org/licenses/>.
*
* Last Modification:
*
* @author nicolas.guelfi
* @date Mon May 06 18:10:54 CEST 2013
*/

package lu.uni.lassy.messir.libraries.string{

import lu.uni.lassy.messir.libraries.primitives
import lu.uni.lassy.messir.libraries.math
```

```

 Concept Model {

  Primary Types {

   datatype dtString {
    attribute value:ptString

    // Type checking Operation
    operation is():ptBoolean
    external operation close() : ptBoolean

    operation length() : dtInteger
    operation dtStringConcat(AdtString:dtString) : dtString
    operation subdtString(StartIndex:dtInteger,
               EndIndex:dtInteger
                ) : dtString

    operation toLower():dtString
    operation toUpper():dtString

    operation eq(AdtString:dtString):ptBoolean
    operation neq(AdtString:dtString):ptBoolean
    operation geq(AdtString:dtString) : ptBoolean
    operation leq(AdtString:dtString) : ptBoolean
    operation lt(AdtString:dtString) : ptBoolean
    operation gt(AdtString:dtString) : ptBoolean

    // Conversion Operations
    operation toptString():ptString
    }
  }
 }
}
```

Listing B.6: Messir Spec. file string.msr.