

Schema documentation for ana.xsd

september 7, 2012

Table of Contents

Resource hierarchy:	1
Namespace: "http://www.ime.iiasa.ac.at/model/analysis"	1
Schema(s)	1
Main schema ana.xsd	1
Complex Type(s)	2
Complex Type a:analysis	2
Complex Type a:preference	2
Complex Type a:task	2
Simple Type(s)	3
Simple Type a:preferenceType	3
Namespace: "http://www.ime.iiasa.ac.at/model/instance"	4
Schema(s)	4
Imported schema data.xsd	4
Element(s)	4
Element i:dataSet	4
Complex Type(s)	4
Complex Type i:modelData	4
Complex Type i:modelInstance	5
Complex Type i:memberDic	5
Complex Type i:tupleValue	5
Complex Type i:setMembers	6
Complex Type i:tupleMembers	6
Complex Type i:entityValues	6
Complex Type i:tupleValues	7
Complex Type i:value	7
Complex Type i:groupEntityValues	7
Namespace: "http://www.ime.iiasa.ac.at/model/spec"	8
Schema(s)	8
Imported schema sms.xsd	8
Simple Type(s)	8
Simple Type m:shortName	8
Simple Type m:name	8
Simple Type m:description	8
Simple Type m:status	9
Simple Type m:setSpecType	9
Simple Type m:entityRole	9
Simple Type m:mathType	10
Complex Type(s)	10
Complex Type m:modelSpec	10
Complex Type m:setSpec	11
Complex Type m:iteratorContainer	11
Complex Type m:entitySpec	12

Resource hierarchy:

Legend:  Import,  Include,  Redefine,  Cycle detected

ana.xsd

 data.xsd sms.xsd sms.xsd

Namespace: "http://www.ime.iiasa.ac.at/model/analysis"

Schema(s)

Main schema ana.xsd

Namespace	http://www.ime.iiasa.ac.at/model/analysis
-----------	---

Properties	attribute form default:	unqualified
	element form default:	unqualified

Complex Type(s)

Complex Type a:analysis

Namespace	http://www.ime.iiasa.ac.at/model/analysis
Diagram	
Model	id , idModelInstance , shortName , name{0,1} , description{0,1}
Children	description, id, idModelInstance, name, shortName
Source	<pre> <xs:complexType name="analysis"> <xs:sequence> <xs:element name="id" type="xs:int" /> <xs:element name="idModelInstance" type="xs:int" minOccurs="1" maxOccurs="1"/> <xs:element name="shortName" type="m:shortName" /> <xs:element name="name" type="m:name" minOccurs="0" maxOccurs="1"/> <xs:element name="description" type="m:description" minOccurs="0" maxOccurs="1"/> </xs:sequence> </xs:complexType> </pre>

Complex Type a:preference

Namespace	http://www.ime.iiasa.ac.at/model/analysis
Annotations	need to be discussed within the developers group
Diagram	
Model	idEntitySpec , preferenceType , prefValue
Children	idEntitySpec, prefValue, preferenceType
Source	<pre> <xs:complexType name="preference"> <xs:annotation> <xs:documentation>need to be discussed within the developers group</xs:documentation> </xs:annotation> <xs:sequence> <xs:element name="idEntitySpec" type="xs:int" minOccurs="1" maxOccurs="1"/> <xs:element name="preferenceType" type="a:preferenceType" minOccurs="1" maxOccurs="1"/> <xs:element name="prefValue" type="xs:double" minOccurs="1" maxOccurs="1"/> </xs:sequence> </xs:complexType> </pre>

Complex Type a:task

Namespace	http://www.ime.iiasa.ac.at/model/analysis
Annotations	need to be discussed within the developers group

Diagram	
Model	id , idAnalysis , idSolver , name{0,1} , description{0,1} , process
Children	description, id, idAnalysis, idSolver, name, process
Source	<pre> <xs:complexType name="task"> <xs:annotation> <xs:documentation>need to be discussed within the developers group</xs:documentation> </xs:annotation> <xs:sequence> <xs:element name="id" type="xs:int" /> <xs:element name="idAnalysis" type="xs:int" minOccurs="1" maxOccurs="1" /> <xs:element name="idSolver" type="xs:int" minOccurs="1" maxOccurs="1" /> <xs:element name="name" type="m:name" minOccurs="0" maxOccurs="1" /> <xs:element name="description" type="m:description" minOccurs="0" maxOccurs="1" /> <xs:element name="process" type="xs:string" minOccurs="1" maxOccurs="1" /> </xs:sequence> </xs:complexType> </pre>

Simple Type(s)

Simple Type a:preferenceType

Namespace	http://www.ime.iiasa.ac.at/model/analysis								
Annotations	need to be discussed within the developers group								
Diagram									
Type	restriction of xs:string								
Facets	<table> <tr> <td>enumeration</td><td>MIN</td></tr> <tr> <td>enumeration</td><td>MAX</td></tr> <tr> <td>enumeration</td><td>LOWER_BND</td></tr> <tr> <td>enumeration</td><td>UPPER_BND</td></tr> </table>	enumeration	MIN	enumeration	MAX	enumeration	LOWER_BND	enumeration	UPPER_BND
enumeration	MIN								
enumeration	MAX								
enumeration	LOWER_BND								
enumeration	UPPER_BND								
Used by	Element a:preference/preferenceType								
Source	<pre> <xs:simpleType name="preferenceType"> <xs:annotation> <xs:documentation>need to be discussed within the developers group</xs:documentation> </xs:annotation> <xs:restriction base="xs:string"> <xs:enumeration value="MIN" /> <xs:enumeration value="MAX" /> <xs:enumeration value="LOWER_BND" /> <xs:enumeration value="UPPER_BND" /> </xs:restriction> </xs:simpleType> </pre>								

Namespace: "http://www.ime.iiasa.ac.at/model/instance"

Schema(s)

Imported schema data.xsd

Namespace	http://www.ime.iiasa.ac.at/model/instance
Properties	attribute form default: unqualified
	element form default: unqualified

Element(s)

Element i:dataset

Namespace	http://www.ime.iiasa.ac.at/model/instance
Annotations	only for mockup
Diagram	
Properties	content: complex
Model	setMembers+, entityValues+
Children	entityValues, setMembers
Instance	<pre><i:dataset xmlns:i="http://www.ime.iiasa.ac.at/model/instance"> <setMembers>{1,unbounded}</setMembers> <entityValues>{1,unbounded}</entityValues> </i:dataset></pre>
Source	<pre><xs:element name="dataset"> <xs:annotation> <xs:documentation>only for mockup</xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence> <xs:element name="setMembers" type="i:setMembers" minOccurs="1" maxOccurs="unbounded"/> <xs:element name="entityValues" type="i:entityValues" minOccurs="1" maxOccurs="unbounded"/> </xs:sequence> </xs:complexType> </xs:element></pre>

Complex Type(s)

Complex Type i:modelData

Namespace	http://www.ime.iiasa.ac.at/model/instance
Diagram	
Model	id, idModelSpec+, idParent{0,1}, description{0,1}, locked
Children	description, id, idModelSpec, idParent, locked
Source	<pre><xs:complexType name="modelData"> <xs:sequence> <xs:element name="id" type="xs:int"/> <xs:element name="idModelSpec" type="xs:int" minOccurs="1" maxOccurs="unbounded"/> <xs:element name="idParent" type="xs:int" minOccurs="0" maxOccurs="1"/> <xs:element name="description" type="m:description" minOccurs="0" maxOccurs="1"/> <xs:element name="locked" type="xs:boolean" minOccurs="1" maxOccurs="1"/> </xs:sequence> </xs:complexType></pre>

Complex Type i:modelInstance

Namespace	http://www.ime.iiasa.ac.at/model/instance
Diagram	
Model	id , idModelSpec , idModelData , shortName{0,1} , name{0,1} , description{0,1} , status
Children	description, id, idModelData, idModelSpec, name, shortName, status
Source	<pre> <xs:complexType name="modelInstance"> <xs:sequence> <xs:element name="id" type="xs:int"/> <xs:element name="idModelSpec" type="xs:int" minOccurs="1" maxOccurs="1"/> <xs:element name="idModelData" type="xs:int" minOccurs="1" maxOccurs="1"/> <xs:element name="shortName" type="m:shortName" minOccurs="0" maxOccurs="1"/> <xs:element name="name" type="m:name" minOccurs="0" maxOccurs="1"/> <xs:element name="description" type="m:description" minOccurs="0" maxOccurs="1"/> <xs:element name="status" type="m:status" minOccurs="1" maxOccurs="1"/> </xs:sequence> </xs:complexType> </pre>

Complex Type i:memberDic

Namespace	http://www.ime.iiasa.ac.at/model/instance
Diagram	
Used by	Elements i:groupEntityValues/tupleMember, i:tupleValue/tupleMember
Model	id , code , description{0,1}
Children	code, description, id
Source	<pre> <xs:complexType name="memberDic"> <xs:sequence> <xs:element name="id" type="xs:int"/> <xs:element name="code" type="m:name"/> <xs:element name="description" type="m:description" minOccurs="0" maxOccurs="1"/> </xs:sequence> </xs:complexType> </pre>

Complex Type i:tupleValue

Namespace	http://www.ime.iiasa.ac.at/model/instance
Diagram	
Model	tupleMember* , value
Children	tupleMember, value
Source	<pre> <xs:complexType name="tupleValue"> <xs:sequence> <xs:element name="tupleMember" type="i:memberDic" minOccurs="0" maxOccurs="unbounded"/> <xs:element name="value"> <xs:complexType> <xs:choice> <xs:element name="intValue" type="xs:int"/> <xs:element name="doubleValue" type="xs:double"/> </xs:choice> </xs:complexType> </xs:element> </xs:sequence> </xs:complexType> </pre>

```

</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>

```

Complex Type `i:setMembers`

Namespace	http://www.ime.iiasa.ac.at/model/instance
Diagram	
Used by	Element <code>i:dataSet/setMembers</code>
Model	setShortName , tupleMembers*
Children	setShortName, tupleMembers
Source	<pre> <xs:complexType name="setMembers"> <xs:sequence> <xs:element name="setShortName" type="xs:string" /> <xs:element name="tupleMembers" type="i:tupleMembers" minOccurs="0" maxOccurs="unbounded" /> </xs:sequence> </xs:complexType> </pre>

Complex Type `i:tupleMembers`

Namespace	http://www.ime.iiasa.ac.at/model/instance
Diagram	
Used by	Element <code>i:setMembers/tupleMembers</code>
Model	tuple{0,1} , members
Children	members, tuple
Source	<pre> <xs:complexType name="tupleMembers"> <xs:sequence> <xs:element name="tuple" type="xs:string" minOccurs="0" maxOccurs="1"> <xs:annotation> <xs:documentation>only for indexed set, tuple members are seperated by , for example: AT,2007</xs:documentation> </xs:annotation> </xs:element> <xs:element name="members" type="xs:string" minOccurs="1" maxOccurs="1"> <xs:annotation> <xs:documentation>set members are seperated by , for example: AT,AU,PL</xs:documentation> </xs:annotation> </xs:element> </xs:sequence> </xs:complexType> </pre>

Complex Type `i:entityValues`

Namespace	http://www.ime.iiasa.ac.at/model/instance
Annotations	only for mockup
Diagram	
Used by	Element <code>i:dataSet/entityValues</code>
Model	entityShortName , tupleValues*
Children	entityShortName, tupleValues
Source	<pre> <xs:complexType name="entityValues"> <xs:annotation> </pre>

```

<xs:documentation>only for mockup</xs:documentation>
</xs:annotation>
<xs:sequence>
  <xs:element name="entityShortName" type="xs:string"/>
  <xs:element name="tupleValues" type="i:tupleValues" minOccurs="0" maxOccurs="unbounded"/>
</xs:sequence>
</xs:complexType>

```

Complex Type i:tupleValues

Namespace	http://www.ime.iiasa.ac.at/model/instance
Annotations	only for mockup
Diagram	
Used by	Element i:entityValues/tupleValues
Model	tuple{0,1} , value
Children	tuple, value
Source	<pre> <xs:complexType name="tupleValues"> <xs:annotation> <xs:documentation>only for mockup</xs:documentation> </xs:annotation> <xs:sequence> <xs:element name="tuple" type="xs:string" minOccurs="0" maxOccurs="1"> <xs:annotation> <xs:documentation>tuple members are separated by , for example: AT,2007</xs:documentation> </xs:annotation> </xs:element> <xs:element name="value" type="i:value"/> </xs:sequence> </xs:complexType> </pre>

Complex Type i:value

Namespace	http://www.ime.iiasa.ac.at/model/instance
Diagram	
Used by	Elements i:groupEntityValues/value, i:tupleValues/value
Model	intValue doubleValue
Children	doubleValue, intValue
Source	<pre> <xs:complexType name="value"> <xs:choice> <xs:element name="intValue" type="xs:int"/> <xs:element name="doubleValue" type="xs:double"/> </xs:choice> </xs:complexType> </pre>

Complex Type i:groupEntityValues

Namespace	http://www.ime.iiasa.ac.at/model/instance
Diagram	
Model	tupleMember*, value*
Children	tupleMember, value
Source	<pre> <xs:complexType name="groupEntityValues"> <xs:sequence> <xs:element name="tupleMember" type="i:memberDic" minOccurs="0" maxOccurs="unbounded"/> <xs:element name="value" type="i:value" minOccurs="0" maxOccurs="unbounded"/> </xs:sequence> </xs:complexType> </pre>

Namespace: "http://www.ime.iiasa.ac.at/model/spec"

Schema(s)

Imported schema sms.xsd

Namespace	http://www.ime.iiasa.ac.at/model/spec		
Properties	attribute form default:	unqualified	
	element form default:	unqualified	

Simple Type(s)

Simple Type m:shortName

Namespace	http://www.ime.iiasa.ac.at/model/spec		
Diagram			
Type	restriction of xs:string		
Facets	minLength	1	
	maxLength	8	
Used by	Elements	a:analysis/shortName, i:modelInstance/shortName, m:entitySpec/shortName, m:modelSpec/shortName, m:setSpec/shortName	
Source	<pre><xs:simpleType name="shortName"> <xs:restriction base="xs:string"> <xs:minLength value="1"/> <xs:maxLength value="8"/> </xs:restriction> </xs:simpleType></pre>		

Simple Type m:name

Namespace	http://www.ime.iiasa.ac.at/model/spec		
Diagram			
Type	restriction of xs:string		
Facets	minLength	1	
	maxLength	32	
Used by	Elements	a:analysis/name, a:task/name, i:memberDic/code, i:modelInstance/name, m:entitySpec/name, m:modelSpec/name, m:setSpec/name	
Source	<pre><xs:simpleType name="name"> <xs:restriction base="xs:string"> <xs:minLength value="1"/> <xs:maxLength value="32"/> </xs:restriction> </xs:simpleType></pre>		

Simple Type m:description

Namespace	http://www.ime.iiasa.ac.at/model/spec		
Diagram			
Type	restriction of xs:string		
Facets	minLength	1	
	maxLength	512	

Used by	Elements a:analysis/description, a:task/description, i:memberDic/description, i:modelData/description, i:modelInstance/description, m:entitySpec/description, m:modelSpec/description, m:setSpec/description
Source	<pre> <xs:simpleType name="description"> <xs:restriction base="xs:string"> <xs:minLength value="1" /> <xs:maxLength value="512" /> </xs:restriction> </xs:simpleType> </pre>

Simple Type m:status

Namespace	http://www.ime.iiasa.ac.at/model/spec						
Diagram							
Type	restriction of xs:string						
Facets	<table> <tr><td>enumeration</td><td>EDIT</td></tr> <tr><td>enumeration</td><td>TEST</td></tr> <tr><td>enumeration</td><td>COMMITTED</td></tr> </table>	enumeration	EDIT	enumeration	TEST	enumeration	COMMITTED
enumeration	EDIT						
enumeration	TEST						
enumeration	COMMITTED						
Used by	Elements i:modelInstance/status, m:modelSpec/status						
Source	<pre> <xs:simpleType name="status"> <xs:restriction base="xs:string"> <xs:enumeration value="EDIT" /> <xs:enumeration value="TEST" /> <xs:enumeration value="COMMITTED" /> </xs:restriction> </xs:simpleType> </pre>						

Simple Type m:setSpecType

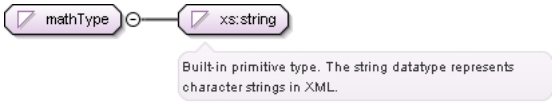
Namespace	http://www.ime.iiasa.ac.at/model/spec				
Diagram					
Type	restriction of xs:string				
Facets	<table> <tr><td>enumeration</td><td>COLLECTION</td></tr> <tr><td>enumeration</td><td>SEQUENCE</td></tr> </table>	enumeration	COLLECTION	enumeration	SEQUENCE
enumeration	COLLECTION				
enumeration	SEQUENCE				
Used by	Element m:setSpec/type				
Source	<pre> <xs:simpleType name="setSpecType"> <xs:restriction base="xs:string"> <xs:enumeration value="COLLECTION" /> <xs:enumeration value="SEQUENCE" /> </xs:restriction> </xs:simpleType> </pre>				

Simple Type m:entityRole

Namespace	http://www.ime.iiasa.ac.at/model/spec												
Diagram													
Type	restriction of xs:string												
Facets	<table> <tr><td>enumeration</td><td>CONSTANT</td></tr> <tr><td>enumeration</td><td>PARAMETER</td></tr> <tr><td>enumeration</td><td>DECISION_VAR</td></tr> <tr><td>enumeration</td><td>EXTERNAL_DECISION_VAR</td></tr> <tr><td>enumeration</td><td>OUTCOME_VAR</td></tr> <tr><td>enumeration</td><td>AUXILIARY_VAR</td></tr> </table>	enumeration	CONSTANT	enumeration	PARAMETER	enumeration	DECISION_VAR	enumeration	EXTERNAL_DECISION_VAR	enumeration	OUTCOME_VAR	enumeration	AUXILIARY_VAR
enumeration	CONSTANT												
enumeration	PARAMETER												
enumeration	DECISION_VAR												
enumeration	EXTERNAL_DECISION_VAR												
enumeration	OUTCOME_VAR												
enumeration	AUXILIARY_VAR												

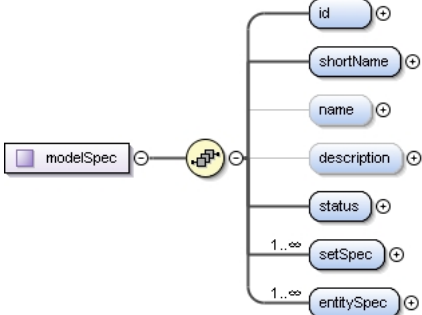
	enumeration	ASSIGNMENT	
	enumeration	CONSTRAINT	
	enumeration	A_VARIABLE	any type of variable: DECISION_VAR, EXTERNAL_DECISION_VAR, OUTCOME_VAR, or AUXILIARY_VAR
	enumeration	A_FORMULA	any type of formula: ASSIGNMENT or CONSTRAINT
Used by	Element	m:entitySpec/role	
Source	<pre> <xs:simpleType name="entityRole"> <xs:restriction base="xs:string"> <xs:enumeration value="CONSTANT" /> <xs:enumeration value="PARAMETER" /> <xs:enumeration value="DECISION_VAR" /> <xs:enumeration value="EXTERNAL_DECISION_VAR" /> <xs:enumeration value="OUTCOME_VAR" /> <xs:enumeration value="AUXILIARY_VAR" /> <xs:enumeration value="ASSIGNMENT" /> <xs:enumeration value="CONSTRAINT" /> <xs:enumeration value="A_VARIABLE"> <xs:annotation> <xs:documentation>any type of variable: DECISION_VAR, EXTERNAL_DECISION_VAR, OUTCOME_VAR, or AUXILIARY_VAR</xs:documentation> </xs:annotation> </xs:enumeration> <xs:enumeration value="A_FORMULA"> <xs:annotation> <xs:documentation>any type of formula: ASSIGNMENT or CONSTRAINT</xs:documentation> </xs:annotation> </xs:enumeration> </xs:restriction> </xs:simpleType> </pre>		

Simple Type m:mathType

Namespace	http://www.ime.iiasa.ac.at/model/spec				
Diagram					
Type	restriction of xs:string				
Facets	<table> <tr> <td>enumeration</td> <td>INTEGER</td> </tr> <tr> <td>enumeration</td> <td>REAL</td> </tr> </table>	enumeration	INTEGER	enumeration	REAL
enumeration	INTEGER				
enumeration	REAL				
Used by	Element m:entitySpec/mathType				
Source	<pre> <xs:simpleType name="mathType"> <xs:restriction base="xs:string"> <xs:enumeration value="INTEGER" /> <!-- <xs:enumeration value="BINARY" /> --> <xs:enumeration value="REAL" /> </xs:restriction> </xs:simpleType> </pre>				

Complex Type(s)

Complex Type m:modelSpec

Namespace	http://www.ime.iiasa.ac.at/model/spec
Diagram	

Model	id , shortName , name{0,1} , description{0,1} , status , setSpec+ , entitySpec+
Children	description, entitySpec, id, name, setSpec, shortName, status
Source	<pre> <xs:complexType name="modelSpec"> <xs:sequence> <xs:element name="id" type="xs:int"/> <xs:element name="shortName" type="m:shortName"/> <xs:element name="name" type="m:name" minOccurs="0" maxOccurs="1"/> <xs:element name="description" type="m:description" minOccurs="0" maxOccurs="1"/> <xs:element name="status" type="m:status"/> <xs:element name="setSpec" type="m:setSpec" maxOccurs="unbounded"/> <xs:element name="entitySpec" type="m:entitySpec" maxOccurs="unbounded"/> </xs:sequence> </xs:complexType> </pre>

Complex Type m:setSpec

Namespace	http://www.ime.iiasa.ac.at/model/spec
Diagram	
Used by	Element m:modelSpec/setSpec
Model	id , idParent{0,1} , shortName , name{0,1} , description{0,1} , type , idx{0,1} , iteratorContainer{0,1}
Children	description, id, idParent, idx, iteratorContainer, name, shortName, type
Source	<pre> <xs:complexType name="setSpec"> <xs:sequence> <xs:element name="id" type="xs:int"/> <xs:element name="idParent" type="xs:int" minOccurs="0" maxOccurs="1"> <xs:annotation> <xs:documentation>only for sub set</xs:documentation> </xs:annotation> </xs:element> <xs:element name="shortName" type="m:shortName"/> <xs:element name="name" type="m:name" minOccurs="0" maxOccurs="1"/> <xs:element name="description" type="m:description" minOccurs="0" maxOccurs="1"/> <xs:element name="type" type="m:setSpecType"/> <xs:element name="idx" type="xs:string" minOccurs="0" maxOccurs="1"> <xs:annotation> <xs:documentation>default index name (for main set only)</xs:documentation> </xs:annotation> </xs:element> <xs:element name="iteratorContainer" type="m:iteratorContainer" minOccurs="0" maxOccurs="1"> <xs:annotation> <xs:documentation>for index set only</xs:documentation> </xs:annotation> </xs:element> </xs:sequence> </xs:complexType> </pre>

Complex Type m:iteratorContainer

Namespace	http://www.ime.iiasa.ac.at/model/spec
Diagram	

Used by	Elements m:entitySpec/iteratorContainer, m:setSpec/iteratorContainer
Model	idSetSpec+
Children	idSetSpec
Source	<pre> <xs:complexType name="iteratorContainer"> <xs:sequence> <xs:element name="idSetSpec" type="xs:int" minOccurs="1" maxOccurs="unbounded"/> </xs:sequence> </xs:complexType> </pre>

Complex Type m:entitySpec

Namespace	http://www.ime.iiasa.ac.at/model/spec
Diagram	
Used by	Element m:modelSpec/entitySpec
Model	id , shortName , name{0,1} , description{0,1} , idLowerBound{0,1} , idUpperBound{0,1} , constantValue{0,1} , iteratorContainer{0,1} , role{0,1} , mathType{0,1} , unit{0,1} , formula{0,1} , source{0,1} , group*, modelSpec{0,1}
Children	constantValue, description, formula, group, id, idLowerBound, idUpperBound, iteratorContainer, mathType, modelSpec, name, role, shortName, source, unit
Source	<pre> <xs:complexType name="entitySpec"> <xs:sequence> <xs:element name="id" type="xs:int"/> <xs:element name="shortName" type="m:shortName"/> <xs:element name="name" type="m:name" minOccurs="0" maxOccurs="1"/> <xs:element name="description" type="m:description" minOccurs="0" maxOccurs="1"/> <xs:element name="idLowerBound" type="xs:int" minOccurs="0" maxOccurs="1"/> <xs:element name="idUpperBound" type="xs:int" minOccurs="0" maxOccurs="1"/> <xs:element name="constantValue" type="xs:double" minOccurs="0" maxOccurs="1"> <xs:annotation> <xs:documentation>optional, only for constant type of entity</xs:documentation> </xs:annotation> </xs:element> <xs:element name="iteratorContainer" type="m:iteratorContainer" minOccurs="0" maxOccurs="1"/> <xs:element name="role" type="m:entityRole" minOccurs="0" maxOccurs="1"/> <xs:element name="mathType" type="m:mathType" minOccurs="0" maxOccurs="1"/> <xs:element name="unit" type="xs:string" minOccurs="0" maxOccurs="1"/> <xs:element name="formula" type="xs:string" minOccurs="0" maxOccurs="1"/> <xs:element name="source" type="xs:string" minOccurs="0" maxOccurs="1"> <xs:annotation> <xs:documentation>specification of the data source (optional, but very useful for organizing data management processes)</xs:documentation> </xs:annotation> </xs:element> <xs:element name="group" type="m:group" minOccurs="0" maxOccurs="∞"> <xs:annotation> <xs:documentation>for grouping parameters (optional, but very useful for management of the data access rights)</xs:documentation> </xs:annotation> </xs:element> <xs:element name="modelSpec" type="m:modelSpec" minOccurs="0" maxOccurs="1"/> </xs:sequence> </xs:complexType> </pre>

```
<xs:annotation>
  <xs:documentation>specification of the data source (optional, but very useful for organizing
data management processes</xs:documentation>
</xs:annotation>
</xs:element>
<xs:element name="group" type="xs:string" minOccurs="0" maxOccurs="unbounded">
  <xs:annotation>
    <xs:documentation>for grouping parameters (optional, but very useful for management of the
data access rights).</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="modelSpec" type="xs:string" minOccurs="0" maxOccurs="1"/>
</xs:sequence>
</xs:complexType>
```