

Schema documentation for ana.xsd

september 19, 2012

Table of Contents

Resource hierarchy:	2
Namespace: "http://www.ime.iiasa.ac.at/model/analysis"	3
Schema(s)	3
Main schema ana.xsd	3
Complex Type(s)	3
Complex Type a:analysis	3
Complex Type a:preference	3
Complex Type a:task	4
Simple Type(s)	4
Simple Type a:preferenceType	4
Namespace: "http://www.ime.iiasa.ac.at/model/instance"	5
Schema(s)	5
Imported schema data.xsd	5
Element(s)	5
Element i:dataSet	5
Complex Type(s)	5
Complex Type i:modelData	5
Complex Type i:modelInstance	6
Complex Type i:memberDic	6
Complex Type i:tupleValue	6
Complex Type i:setMembers	7
Complex Type i:tupleMembers	7
Complex Type i:entityValues	7
Complex Type i:tupleValues	8
Complex Type i:groupEntityValues	8
Namespace: "http://www.ime.iiasa.ac.at/model/spec"	9
Schema(s)	9
Imported schema sms.xsd	9
Simple Type(s)	9
Simple Type m:shortName	9
Simple Type m:name	9
Simple Type m:description	9
Simple Type m:status	10
Simple Type m:setSpecType	10
Simple Type m:entityRole	10
Simple Type m:mathType	11
Complex Type(s)	11
Complex Type m:modelSpec	11
Complex Type m:setSpec	12
Complex Type m:iteratorContainer	13
Complex Type m:entitySpec	13
Namespace: ""	14
Element(s)	14
Element a:analysis / id	14
Element a:analysis / idModelInstance	14
Element a:analysis / shortName	14
Element a:analysis / name	15
Element a:analysis / description	15
Element a:preference / idEntitySpec	15
Element a:preference / preferenceType	15
Element a:preference / prefValue	16
Element a:task / id	16
Element a:task / idAnalysis	16
Element a:task / idSolver	16
Element a:task / name	17
Element a:task / description	17
Element a:task / process	17
Element i:modelData / id	17
Element i:modelData / idModelSpec	18
Element i:modelData / idParent	18
Element i:modelData / description	18
Element i:modelData / locked	18

Element i:modelInstance / id	19
Element i:modelInstance / idModelSpec	19
Element i:modelInstance / idModelData	19
Element i:modelInstance / shortName	19
Element i:modelInstance / name	19
Element i:modelInstance / description	20
Element i:modelInstance / status	20
Element i:memberDic / id	20
Element i:memberDic / code	20
Element i:memberDic / code / stringValue	21
Element i:memberDic / code / intValue	21
Element i:memberDic / code / doubleValue	21
Element i:memberDic / description	21
Element i:tupleValue / tupleMember	22
Element i:tupleValue / value	22
Element i:tupleValue / value / intValue	22
Element i:tupleValue / value / doubleValue	23
Element i:setMembers / setShortName	23
Element i:setMembers / tupleMembers	23
Element i:tupleMembers / tuple	23
Element i:tupleMembers / members	24
Element i:dataSet / setMembers	24
Element i:dataSet / entityValues	24
Element i:entityValues / entityShortName	25
Element i:entityValues / tupleValues	25
Element i:tupleValues / tuple	25
Element i:tupleValues / value	26
Element i:tupleValues / value / intValue	26
Element i:tupleValues / value / doubleValue	26
Element i:groupEntityValues / tupleMember	26
Element i:groupEntityValues / value	27
Element i:groupEntityValues / value / intValue	27
Element i:groupEntityValues / value / doubleValue	27
Element m:modelSpec / id	28
Element m:modelSpec / shortName	28
Element m:modelSpec / name	28
Element m:modelSpec / description	28
Element m:modelSpec / status	28
Element m:modelSpec / setSpec	29
Element m:setSpec / id	29
Element m:setSpec / idParent	29
Element m:setSpec / shortName	30
Element m:setSpec / name	30
Element m:setSpec / description	30
Element m:setSpec / type	30
Element m:setSpec / idx	31
Element m:setSpec / iteratorContainer	31
Element m:iteratorContainer / idSetSpec	31
Element m:modelSpec / entitySpec	32
Element m:entitySpec / id	33
Element m:entitySpec / shortName	33
Element m:entitySpec / name	33
Element m:entitySpec / description	33
Element m:entitySpec / idLowerBound	33
Element m:entitySpec / idUpperBound	34
Element m:entitySpec / constantValue	34
Element m:entitySpec / iteratorContainer	34
Element m:entitySpec / role	35
Element m:entitySpec / mathType	35
Element m:entitySpec / unit	35
Element m:entitySpec / formula	36
Element m:entitySpec / source	36
Element m:entitySpec / group	36
Element m:entitySpec / modelSpec	36

Resource hierarchy:

Legend:  Import,  Include,  Redefine,  Cycle detected

ana.xsd

 data.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></pre>

```

<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>

```

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:restriction> </xs:simpleType> </pre>								

```

<xs:enumeration value="UPPER_BND" />
</xs:restriction>
</xs:simpleType>

```

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:sequence> </xs:complexType> </pre>

```

<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>

```

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"> <xs:complexType> <xs:choice> <xs:element name="stringValue" type="xs:string"/> <xs:element name="intValue" type="xs:int"/> <xs:element name="doubleValue" type="xs:double"/> </xs:choice> </xs:complexType> </xs:element> <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>

Complex Type i:setMembers

Namespace	http://www.ime.iiasa.ac.at/model/instance
Diagram	
Used by	Element i:dataSet/setMembers
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 i:setMembers/tupleMembers
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 separated 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 separated 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> <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> </pre>

Complex Type `i:tupleValues`

Namespace	http://www.ime.iiasa.ac.at/model/instance
Annotations	only for mockup
Diagram	
Used by	Element <code>i:entityValues/tupleValues</code>
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 seperated by , for example: AT,2007</xs:documentation> </xs:annotation> </xs:element> <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>

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" minOccurs="0" maxOccurs="unbounded"> <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:choice>
      </xs:complexType>
    </xs:element>
  </xs:sequence>
</xs:complexType>

```

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

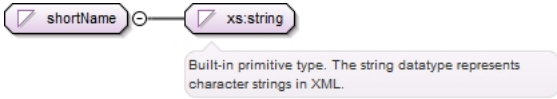
Schema(s)

Imported schema **sms.xsd**

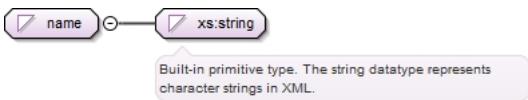
Namespace	http://www.ime.iiasa.ac.at/model/spec
Properties	attribute form default: unqualified
	element form default: unqualified

Simple Type(s)

Simple Type m:shortName

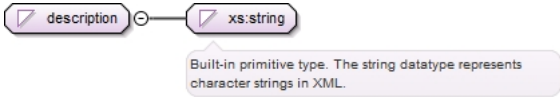
Namespace	http://www.ime.iiasa.ac.at/model/spec
Diagram	
Type	restriction of xs:string
Facets	minLength 1
	maxLength 8
Used by	Elements a:analysis/shortName, i:modelInstance/shortName, m:entitySpec/shortName, m:modelSpec/shortName, m:setSpec/shortName
Source	<pre> <xs:simpleType name="shortName"> <xs:restriction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

Simple Type m:name

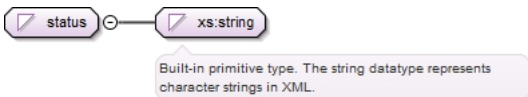
Namespace	http://www.ime.iiasa.ac.at/model/spec
Diagram	
Type	restriction of xs:string
Facets	minLength 1
	maxLength 32
Used by	Elements a:analysis/name, a:task/name, i:modelInstance/name, m:entitySpec/name, m:modelSpec/name, m:setSpec/name
Source	<pre> <xs:simpleType name="name"> <xs:restriction base="xs:string">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

Simple Type m:description

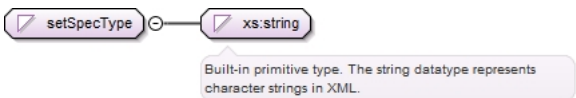
Namespace	http://www.ime.iiasa.ac.at/model/spec
-----------	---------------------------------------

Diagram					
Type	restriction of xs:string				
Facets	<table> <tr> <td>minLength</td><td>1</td></tr> <tr> <td>maxLength</td><td>512</td></tr> </table>	minLength	1	maxLength	512
minLength	1				
maxLength	512				
Used by	<table> <tr> <td>Elements</td><td>a:analysis/description, a:task/description, i:memberDic/description, i:modelData/description, i:modelInstance/description, m:entitySpec/description, m:modelSpec/description, m:setSpec/description</td></tr> </table>	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		
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>				

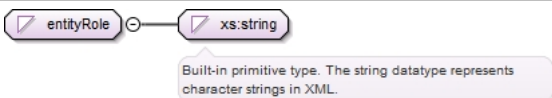
Simple Type **m:status**

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	<table> <tr> <td>Elements</td><td>i:modelInstance/status, m:modelSpec/status</td></tr> </table>	Elements	i:modelInstance/status, m:modelSpec/status				
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	<table> <tr> <td>Element</td><td>m:setSpec/type</td></tr> </table>	Element	m:setSpec/type		
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	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	enumeration	INTEGER	
	enumeration	BINARY	
	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>

```

<xs:documentation>for index set only</xs:documentation>
</xs:annotation>
</xs:element>
</xs:sequence>
</xs:complexType>

```

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" /> </pre>

```

<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="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>

```

Namespace: ""

Element(s)

Element a:analysis / id

Namespace	No namespace
Diagram	
Type	xs:int
Properties	content: simple
Source	<code><xs:element name="id" type="xs:int"/></code>

Element a:analysis / idModelInstance

Namespace	No namespace
Diagram	
Type	xs:int
Properties	content: simple minOccurs: 1 maxOccurs: 1
Source	<code><xs:element name="idModelInstance" type="xs:int" minOccurs="1" maxOccurs="1"/></code>

Element a:analysis / shortName

Namespace	No namespace
Diagram	
Type	m:shortName
Properties	content: simple

Facets	minLength	1
	maxLength	8
Source	<xs:element name="shortName" type="m:shortName"/>	

Element a:analysis / name

Namespace	No namespace	
Diagram		
Type	m:name	
Properties	content:	simple
	minOccurs:	0
	maxOccurs:	1
Facets	minLength	1
	maxLength	32
Source	<xs:element name="name" type="m:name" minOccurs="0" maxOccurs="1"/>	

Element a:analysis / description

Namespace	No namespace	
Diagram		
Type	m:description	
Properties	content:	simple
	minOccurs:	0
	maxOccurs:	1
Facets	minLength	1
	maxLength	512
Source	<xs:element name="description" type="m:description" minOccurs="0" maxOccurs="1"/>	

Element a:preference / idEntitySpec

Namespace	No namespace	
Diagram		
Type	xs:int	
Properties	content:	simple
	minOccurs:	1
	maxOccurs:	1
Source	<xs:element name="idEntitySpec" type="xs:int" minOccurs="1" maxOccurs="1"/>	

Element a:preference / preferenceType

Namespace	No namespace	
Diagram		
Type	a:preferenceType	
Properties	content:	simple

Facets	minOccurs:	1
	maxOccurs:	1
	enumeration	MIN
	enumeration	MAX
	enumeration	LOWER_BND
Facets	enumeration	UPPER_BND
	enumeration	
	enumeration	
	enumeration	
Source	<code><xs:element name="preferenceType" type="a:preferenceType" minOccurs="1" maxOccurs="1"/></code>	

Element a:preference / prefValue

Namespace	No namespace
Diagram	
Type	xs:double
Properties	content: simple
	minOccurs: 1
	maxOccurs: 1
Source	<code><xs:element name="prefValue" type="xs:double" minOccurs="1" maxOccurs="1"/></code>

Element a:task / id

Namespace	No namespace
Diagram	
Type	xs:int
Properties	content: simple
Source	<code><xs:element name="id" type="xs:int"/></code>

Element a:task / idAnalysis

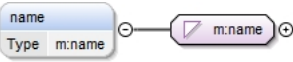
Namespace	No namespace
Diagram	
Type	xs:int
Properties	content: simple
	minOccurs: 1
	maxOccurs: 1
Source	<code><xs:element name="idAnalysis" type="xs:int" minOccurs="1" maxOccurs="1"/></code>

Element a:task / idSolver

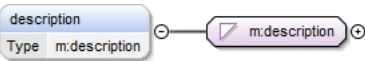
Namespace	No namespace
Diagram	
Type	xs:int

Properties	content:	simple
	minOccurs:	1
	maxOccurs:	1
Source	<xs:element name="idSolver" type="xs:int" minOccurs="1" maxOccurs="1"/>	

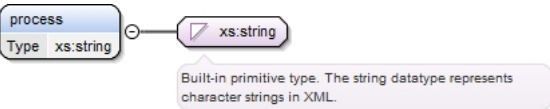
Element a:task / name

Namespace	No namespace	
Diagram		
Type	m:name	
Properties	content:	simple
	minOccurs:	0
	maxOccurs:	1
Facets	minLength	1
	maxLength	32
Source	<xs:element name="name" type="m:name" minOccurs="0" maxOccurs="1"/>	

Element a:task / description

Namespace	No namespace	
Diagram		
Type	m:description	
Properties	content:	simple
	minOccurs:	0
	maxOccurs:	1
Facets	minLength	1
	maxLength	512
Source	<xs:element name="description" type="m:description" minOccurs="0" maxOccurs="1"/>	

Element a:task / process

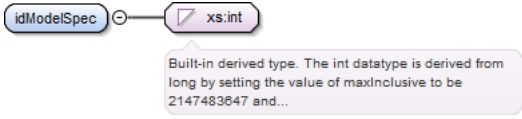
Namespace	No namespace	
Diagram		
Type	xs:string	
Properties	content:	simple
	minOccurs:	1
	maxOccurs:	1
Source	<xs:element name="process" type="xs:string" minOccurs="1" maxOccurs="1"/>	

Element i:modelData / id

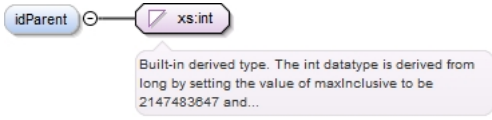
Namespace	No namespace	
Diagram		
Type	xs:int	

Properties	content: simple
Source	<code><xs:element name="id" type="xs:int"/></code>

Element i:modelData / idModelSpec

Namespace	No namespace
Diagram	 <p>The diagram shows an element named 'idModelSpec' connected to a base type 'xs:int'. A callout box for 'xs:int' states: 'Built-in derived type. The int datatype is derived from long by setting the value of maxInclusive to be 2147483647 and...'</p>
Type	xs:int
Properties	content: simple minOccurs: 1 maxOccurs: unbounded
Source	<code><xs:element name="idModelSpec" type="xs:int" minOccurs="1" maxOccurs="unbounded"/></code>

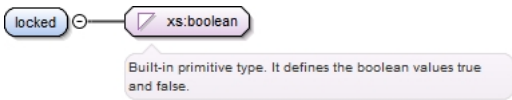
Element i:modelData / idParent

Namespace	No namespace
Diagram	 <p>The diagram shows an element named 'idParent' connected to a base type 'xs:int'. A callout box for 'xs:int' states: 'Built-in derived type. The int datatype is derived from long by setting the value of maxInclusive to be 2147483647 and...'</p>
Type	xs:int
Properties	content: simple minOccurs: 0 maxOccurs: 1
Source	<code><xs:element name="idParent" type="xs:int" minOccurs="0" maxOccurs="1"/></code>

Element i:modelData / description

Namespace	No namespace
Diagram	 <p>The diagram shows an element named 'description' connected to a base type 'm:description'. A callout box for 'm:description' contains a plus sign icon.</p>
Type	m:description
Properties	content: simple minOccurs: 0 maxOccurs: 1
Facets	minLength 1 maxLength 512
Source	<code><xs:element name="description" type="m:description" minOccurs="0" maxOccurs="1"/></code>

Element i:modelData / locked

Namespace	No namespace
Diagram	 <p>The diagram shows an element named 'locked' connected to a base type 'xs:boolean'. A callout box for 'xs:boolean' states: 'Built-in primitive type. It defines the boolean values true and false.'</p>
Type	xs:boolean
Properties	content: simple minOccurs: 1 maxOccurs: 1
Source	<code><xs:element name="locked" type="xs:boolean" minOccurs="1" maxOccurs="1"/></code>

Element i:modelInstance / id

Namespace	No namespace
Diagram	
Type	xs:int
Properties	content: simple
Source	<code><xs:element name="id" type="xs:int"/></code>

Element i:modelInstance / idModelSpec

Namespace	No namespace
Diagram	
Type	xs:int
Properties	content: simple minOccurs: 1 maxOccurs: 1
Source	<code><xs:element name="idModelSpec" type="xs:int" minOccurs="1" maxOccurs="1"/></code>

Element i:modelInstance / idModelData

Namespace	No namespace
Diagram	
Type	xs:int
Properties	content: simple minOccurs: 1 maxOccurs: 1
Source	<code><xs:element name="idModelData" type="xs:int" minOccurs="1" maxOccurs="1"/></code>

Element i:modelInstance / shortName

Namespace	No namespace
Diagram	
Type	m:shortName
Properties	content: simple minOccurs: 0 maxOccurs: 1
Facets	minLength 1 maxLength 8
Source	<code><xs:element name="shortName" type="m:shortName" minOccurs="0" maxOccurs="1"/></code>

Element i:modelInstance / name


Namespace	No namespace
Diagram	

Type	m:name	
Properties	content:	simple
	minOccurs:	0
	maxOccurs:	1
Facets	minLength	1
	maxLength	32
Source	<code><xs:element name="name" type="m:name" minOccurs="0" maxOccurs="1"/></code>	

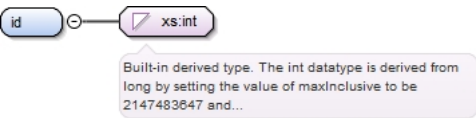
Element i:modelInstance / description

Namespace	No namespace	
Diagram		
Type	m:description	
Properties	content:	simple
	minOccurs:	0
	maxOccurs:	1
Facets	minLength	1
	maxLength	512
Source	<code><xs:element name="description" type="m:description" minOccurs="0" maxOccurs="1"/></code>	

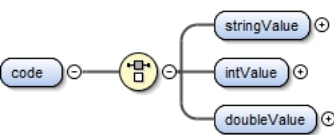
Element i:modelInstance / status

Namespace	No namespace	
Diagram		
Type	m:status	
Properties	content:	simple
	minOccurs:	1
	maxOccurs:	1
Facets	enumeration	EDIT
	enumeration	TEST
	enumeration	COMMITTED
Source	<code><xs:element name="status" type="m:status" minOccurs="1" maxOccurs="1"/></code>	

Element i:memberDic / id

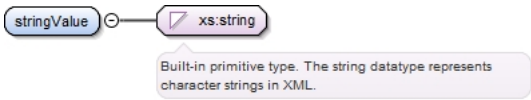
Namespace	No namespace	
Diagram		
Type	xs:int	
Properties	content:	simple
Source	<code><xs:element name="id" type="xs:int"/></code>	

Element i:memberDic / code

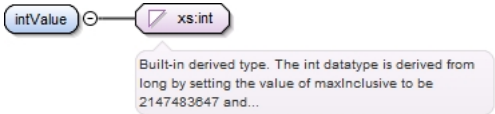
Namespace	No namespace	
Diagram		

Properties	content: complex
Model	stringValue intValue doubleValue
Children	doubleValue, intValue, stringValue
Instance	<pre><code> <stringValue>{1,1}</stringValue> <intValue>{1,1}</intValue> <doubleValue>{1,1}</doubleValue> </code></pre>
Source	<pre><xs:element name="code"> <xs:complexType> <xs:choice> <xs:element name="stringValue" type="xs:string"/> <xs:element name="intValue" type="xs:int"/> <xs:element name="doubleValue" type="xs:double"/> </xs:choice> </xs:complexType> </xs:element></pre>

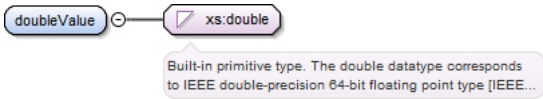
Element i:memberDic / code / stringValue

Namespace	No namespace
Diagram	
Type	xs:string
Properties	content: simple
Source	<pre><xs:element name="stringValue" type="xs:string"/></pre>


Element i:memberDic / code / intValue

Namespace	No namespace
Diagram	
Type	xs:int
Properties	content: simple
Source	<pre><xs:element name="intValue" type="xs:int"/></pre>

Element i:memberDic / code / doubleValue

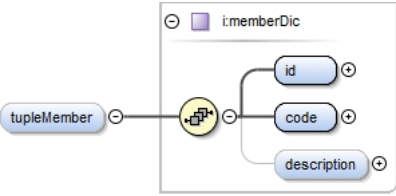
Namespace	No namespace
Diagram	
Type	xs:double
Properties	content: simple
Source	<pre><xs:element name="doubleValue" type="xs:double"/></pre>

Element i:memberDic / description

Namespace	No namespace						
Diagram							
Type	m:description						
Properties	<table> <tr> <td>content:</td><td>simple</td></tr> <tr> <td>minOccurs:</td><td>0</td></tr> <tr> <td>maxOccurs:</td><td>1</td></tr> </table>	content:	simple	minOccurs:	0	maxOccurs:	1
content:	simple						
minOccurs:	0						
maxOccurs:	1						

Facets	minLength	1
	maxLength	512
Source	<code><xs:element name="description" type="m:description" minOccurs="0" maxOccurs="1"/></code>	

Element i:tupleValue / tupleMember

Namespace	No namespace
Diagram	 <p>The diagram illustrates the structure of the <code>tupleMember</code> element. It is connected to a container labeled <code>i:memberDic</code>. Inside this container, there are three child elements: <code>id</code>, <code>code</code>, and <code>description</code>.</p>
Type	i:memberDic
Properties	content: complex
	minOccurs: 0
	maxOccurs: unbounded
Model	id , code , description{0,1}
Children	code, description, id
Instance	<pre><tupleMember> <id>{1,1}</id> <code>{1,1}</code> <description>{0,1}</description> </tupleMember></pre>
Source	<pre><xs:element name="tupleMember" type="i:memberDic" minOccurs="0" maxOccurs="unbounded" /></pre>

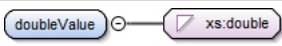
Element i:tupleValue / value

Namespace	No namespace
Diagram	
Properties	content: complex
Model	intValue doubleValue
Children	doubleValue, intValue
Instance	<pre><value> <intValue>{1,1}</intValue> <doubleValue>{1,1}</doubleValue> </value></pre>
Source	<pre><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></pre>

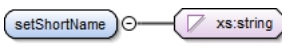
Element i:tupleValue / value / intValue

Namespace	No namespace
Diagram	
Type	xs:int
Properties	content: simple
Source	<xs:element name="intValue" type="xs:int" />

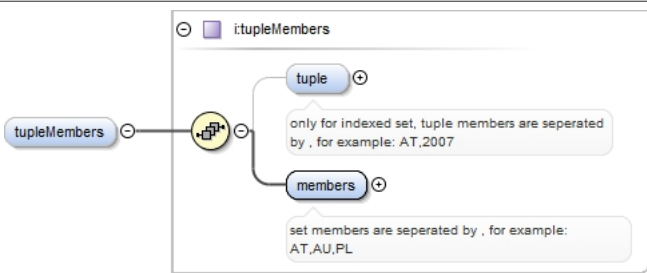
Element i:tupleValue / value / doubleValue

Namespace	No namespace
Diagram	 <p>Built-in primitive type. The double datatype corresponds to IEEE double-precision 64-bit floating point type [IEEE...</p>
Type	xs:double
Properties	content: simple
Source	<code><xs:element name="doubleValue" type="xs:double" /></code>

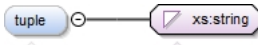
Element i:setMembers / setShortName

Namespace	No namespace
Diagram	 <p>Built-in primitive type. The string datatype represents character strings in XML.</p>
Type	xs:string
Properties	content: simple
Source	<code><xs:element name="setShortName" type="xs:string" /></code>

Element i:setMembers / tupleMembers

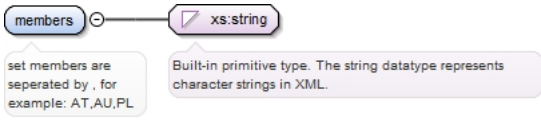
Namespace	No namespace
Diagram	 <p>only for indexed set, tuple members are separated by , for example: AT,2007</p> <p>set members are separated by , for example: AT,AU,PL</p>
Type	i:tupleMembers
Properties	content: complex minOccurs: 0 maxOccurs: unbounded
Model	tuple{0,1} , members
Children	members, tuple
Instance	<pre><tupleMembers> <tuple>{0,1}</tuple> <members>{1,1}</members> </tupleMembers></pre>
Source	<code><xs:element name="tupleMembers" type="i:tupleMembers" minOccurs="0" maxOccurs="unbounded" /></code>

Element i:tupleMembers / tuple

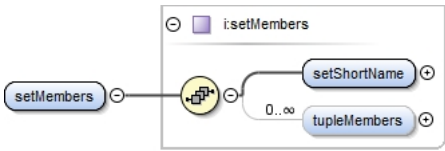
Namespace	No namespace
Annotations	only for indexed set, tuple members are separated by , for example: AT,2007
Diagram	 <p>only for indexed set, tuple members are separated by , for example: AT,2007</p> <p>Built-in primitive type. The string datatype represents character strings in XML.</p>
Type	xs:string

Properties	content:	simple
	minOccurs:	0
	maxOccurs:	1
Source	<pre><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></pre>	

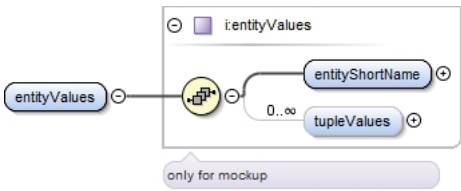
Element i:tupleMembers / members

Namespace	No namespace	
Annotations	set members are seperated by , for example: AT,AU,PL	
Diagram		
Type	xs:string	
Properties	content:	simple
	minOccurs:	1
	maxOccurs:	1
Source	<pre><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></pre>	

Element i:dataset / setMembers

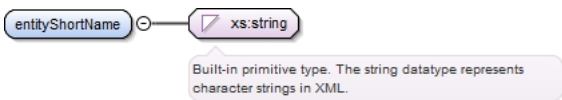
Namespace	No namespace	
Diagram		
Type	i:setMembers	
Properties	content:	complex
	minOccurs:	1
	maxOccurs:	unbounded
Model	setShortName , tupleMembers*	
Children	setShortName, tupleMembers	
Instance	<pre><setMembers> <setShortName>{1,1}</setShortName> <tupleMembers>{0,unbounded}</tupleMembers> </setMembers></pre>	
Source	<pre><xs:element name="setMembers" type="i:setMembers" minOccurs="1" maxOccurs="unbounded"/></pre>	

Element i:dataset / entityValues

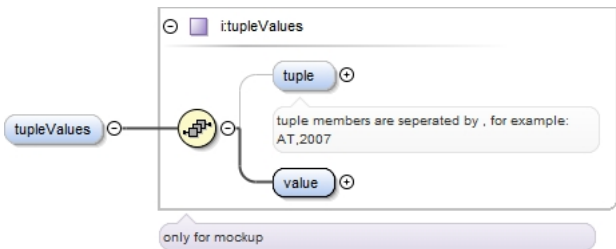
Namespace	No namespace	
Diagram		
Type	i:entityValues	

Properties	content:	complex
	minOccurs:	1
	maxOccurs:	unbounded
Model	entityShortName , tupleValues*	
Children	entityShortName, tupleValues	
Instance	<pre><entityValues> <entityShortName>{1,1}</entityShortName> <tupleValues>{0,unbounded}</tupleValues> </entityValues></pre>	
Source	<pre><xs:element name="entityValues" type="i:entityValues" minOccurs="1" maxOccurs="unbounded"/></pre>	

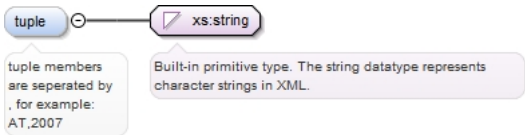
Element i:entityValues / entityShortName

Namespace	No namespace	
Diagram		
Type	xs:string	
Properties	content:	simple
Source	<pre><xs:element name="entityShortName" type="xs:string"/></pre>	

Element i:entityValues / tupleValues

Namespace	No namespace	
Diagram		
Type	i:tupleValues	
Properties	content:	complex
	minOccurs:	0
	maxOccurs:	unbounded
Model	tuple{0,1} , value	
Children	tuple, value	
Instance	<pre><tupleValues> <tuple>{0,1}</tuple> <value>{1,1}</value> </tupleValues></pre>	
Source	<pre><xs:element name="tupleValues" type="i:tupleValues" minOccurs="0" maxOccurs="unbounded"/></pre>	

Element i:tupleValues / tuple

Namespace	No namespace	
Annotations	tuple members are separated by , for example: AT,2007	
Diagram		
Type	xs:string	
Properties	content:	simple
	minOccurs:	0

	maxOccurs: 1
Source	<pre><xs:element name="tuple" type="xs:string" minOccurs="0" maxOccurs="1"> <xs:annotation> <xs:documentation>tuple members are seperated by , for example: AT,2007</xs:documentation> </xs:annotation> </xs:element></pre>

Element i:tupleValues / value

Namespace	No namespace
Diagram	
Properties	content: complex
Model	intValue doubleValue
Children	doubleValue, intValue
Instance	<pre><value> <intValue>{1,1}</intValue> <doubleValue>{1,1}</doubleValue> </value></pre>
Source	<pre><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></pre>

Element i:tupleValues / value / intValue

Namespace	No namespace
Diagram	
Type	xs:int
Properties	content: simple
Source	<pre><xs:element name="intValue" type="xs:int"/></pre>

Element i:tupleValues / value / doubleValue

Namespace	No namespace
Diagram	
Type	xs:double
Properties	content: simple
Source	<pre><xs:element name="doubleValue" type="xs:double"/></pre>

Element i:groupEntityValues / tupleMember

Namespace	No namespace
Diagram	

Type	i:memberDic
Properties	content: complex
	minOccurs: 0
	maxOccurs: unbounded
Model	id , code , description{0,1}
Children	code, description, id
Instance	<pre><tupleMember> <id>{1,1}</id> <code>{1,1}</code> <description>{0,1}</description> </tupleMember></pre>
Source	<code><xs:element name="tupleMember" type="i:memberDic" minOccurs="0" maxOccurs="unbounded"/></code>

Element i:groupEntityValues / value

Namespace	No namespace
Diagram	
Properties	content: complex
	minOccurs: 0
	maxOccurs: unbounded
Model	intValue doubleValue
Children	doubleValue, intValue
Instance	<pre><value> <intValue>{1,1}</intValue> <doubleValue>{1,1}</doubleValue> </value></pre>
Source	<pre><xs:element name="value" minOccurs="0" maxOccurs="unbounded"> <xs:complexType> <xs:choice> <xs:element name="intValue" type="xs:int" /> <xs:element name="doubleValue" type="xs:double" /> </xs:choice> </xs:complexType> </xs:element></pre>

Element i:groupEntityValues / value / intValue

Namespace	No namespace
Diagram	
Type	xs:int
Properties	content: simple
Source	<code><xs:element name="intValue" type="xs:int"/></code>

Element i:groupEntityValues / value / doubleValue

Namespace	No namespace
Diagram	
Type	xs:double
Properties	content: simple
Source	<code><xs:element name="doubleValue" type="xs:double"/></code>

Element `m:modelSpec` / `id`

Namespace	No namespace
Diagram	
Type	xs:int
Properties	content: simple
Source	<code><xs:element name="id" type="xs:int"/></code>

Element `m:modelSpec` / `shortName`

Namespace	No namespace				
Diagram					
Type	m:shortName				
Properties	content: simple				
Facets	<table> <tr> <td>minLength</td><td>1</td></tr> <tr> <td>maxLength</td><td>8</td></tr> </table>	minLength	1	maxLength	8
minLength	1				
maxLength	8				
Source	<code><xs:element name="shortName" type="m:shortName"/></code>				

Element `m:modelSpec` / `name`

Namespace	No namespace						
Diagram							
Type	m:name						
Properties	<table> <tr> <td>content:</td><td>simple</td></tr> <tr> <td>minOccurs:</td><td>0</td></tr> <tr> <td>maxOccurs:</td><td>1</td></tr> </table>	content:	simple	minOccurs:	0	maxOccurs:	1
content:	simple						
minOccurs:	0						
maxOccurs:	1						
Facets	<table> <tr> <td>minLength</td><td>1</td></tr> <tr> <td>maxLength</td><td>32</td></tr> </table>	minLength	1	maxLength	32		
minLength	1						
maxLength	32						
Source	<code><xs:element name="name" type="m:name" minOccurs="0" maxOccurs="1"/></code>						

Element `m:modelSpec` / `description`

Namespace	No namespace						
Diagram							
Type	m:description						
Properties	<table> <tr> <td>content:</td><td>simple</td></tr> <tr> <td>minOccurs:</td><td>0</td></tr> <tr> <td>maxOccurs:</td><td>1</td></tr> </table>	content:	simple	minOccurs:	0	maxOccurs:	1
content:	simple						
minOccurs:	0						
maxOccurs:	1						
Facets	<table> <tr> <td>minLength</td><td>1</td></tr> <tr> <td>maxLength</td><td>512</td></tr> </table>	minLength	1	maxLength	512		
minLength	1						
maxLength	512						
Source	<code><xs:element name="description" type="m:description" minOccurs="0" maxOccurs="1"/></code>						

Element `m:modelSpec` / `status`

Namespace	No namespace
Diagram	
Type	m:status

Properties	content:	simple
Facets	enumeration	EDIT
	enumeration	TEST
	enumeration	COMMITTED
Source	<code><xs:element name="status" type="m:status"/></code>	

Element m:modelSpec / setSpec

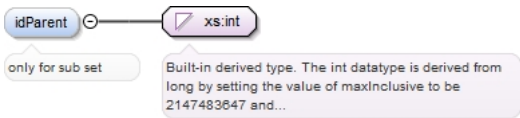
Namespace	No namespace	
Diagram		
Type	m:setSpec	
Properties	content:	complex
	maxOccurs:	unbounded
Model	id , idParent{0,1} , shortName , name{0,1} , description{0,1} , type , idx{0,1} , iteratorContainer{0,1}	
Children	description, id, idParent, idx, iteratorContainer, name, shortName, type	
Instance	<pre> <setSpec> <id>{1,1}</id> <idParent>{0,1}</idParent> <shortName>{1,1}</shortName> <name>{0,1}</name> <description>{0,1}</description> <type>{1,1}</type> <idx>{0,1}</idx> <iteratorContainer>{0,1}</iteratorContainer> </setSpec> </pre>	
Source	<code><xs:element name="setSpec" type="m:setSpec" maxOccurs="unbounded"/></code>	

Element m:setSpec / id


Namespace	No namespace	
Diagram		
Type	xs:int	
Properties	content:	simple
Source	<code><xs:element name="id" type="xs:int"/></code>	

Element m:setSpec / idParent


Namespace	No namespace	
-----------	--------------	--

Annotations	only for sub set						
Diagram							
Type	xs:int						
Properties	<table> <tr><td>content:</td><td>simple</td></tr> <tr><td>minOccurs:</td><td>0</td></tr> <tr><td>maxOccurs:</td><td>1</td></tr> </table>	content:	simple	minOccurs:	0	maxOccurs:	1
content:	simple						
minOccurs:	0						
maxOccurs:	1						
Source	<pre><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></pre>						


Element m:setSpec / shortName

Namespace	No namespace				
Diagram					
Type	m:shortName				
Properties	<table> <tr><td>content:</td><td>simple</td></tr> </table>	content:	simple		
content:	simple				
Facets	<table> <tr><td>minLength</td><td>1</td></tr> <tr><td>maxLength</td><td>8</td></tr> </table>	minLength	1	maxLength	8
minLength	1				
maxLength	8				
Source	<pre><xs:element name="shortName" type="m:shortName" /></pre>				

Element m:setSpec / name


Namespace	No namespace						
Diagram							
Type	m:name						
Properties	<table> <tr><td>content:</td><td>simple</td></tr> <tr><td>minOccurs:</td><td>0</td></tr> <tr><td>maxOccurs:</td><td>1</td></tr> </table>	content:	simple	minOccurs:	0	maxOccurs:	1
content:	simple						
minOccurs:	0						
maxOccurs:	1						
Facets	<table> <tr><td>minLength</td><td>1</td></tr> <tr><td>maxLength</td><td>32</td></tr> </table>	minLength	1	maxLength	32		
minLength	1						
maxLength	32						
Source	<pre><xs:element name="name" type="m:name" minOccurs="0" maxOccurs="1" /></pre>						

Element m:setSpec / description

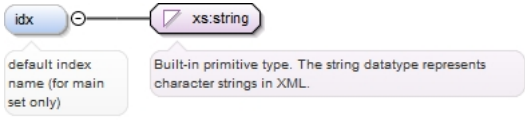
Namespace	No namespace						
Diagram							
Type	m:description						
Properties	<table> <tr><td>content:</td><td>simple</td></tr> <tr><td>minOccurs:</td><td>0</td></tr> <tr><td>maxOccurs:</td><td>1</td></tr> </table>	content:	simple	minOccurs:	0	maxOccurs:	1
content:	simple						
minOccurs:	0						
maxOccurs:	1						
Facets	<table> <tr><td>minLength</td><td>1</td></tr> <tr><td>maxLength</td><td>512</td></tr> </table>	minLength	1	maxLength	512		
minLength	1						
maxLength	512						
Source	<pre><xs:element name="description" type="m:description" minOccurs="0" maxOccurs="1" /></pre>						

Element m:setSpec / type

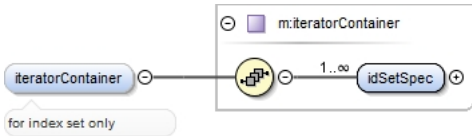
Namespace	No namespace
-----------	--------------

Diagram	
Type	m:setSpecType
Properties	content: simple
Facets	enumeration COLLECTION enumeration SEQUENCE
Source	<pre><xs:element name="type" type="m:setSpecType" /></pre>

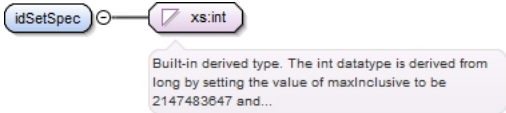
Element m:setSpec / idx

Namespace	No namespace
Annotations	default index name (for main set only)
Diagram	
Type	xs:string
Properties	content: simple minOccurs: 0 maxOccurs: 1
Source	<pre><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></pre>

Element m:setSpec / iteratorContainer

Namespace	No namespace
Annotations	for index set only
Diagram	
Type	m:iteratorContainer
Properties	content: complex minOccurs: 0 maxOccurs: 1
Model	idSetSpec+
Children	idSetSpec
Instance	<pre><iteratorContainer> <idSetSpec>{1,unbounded}</idSetSpec> </iteratorContainer></pre>
Source	<pre><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></pre>

Element m:iteratorContainer / idSetSpec

Namespace	No namespace
Diagram	

Type	xs:int
Properties	content: simple
	minOccurs: 1
	maxOccurs: unbounded
Source	<code><xs:element name="idSetSpec" type="xs:int" minOccurs="1" maxOccurs="unbounded" /></code>

Element m:modelSpec / entitySpec

Namespace	No namespace
Diagram	
Type	m:entitySpec
Properties	content: complex
	maxOccurs: unbounded
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
Instance	<pre> <entitySpec> <id>{1,1}</id> <shortName>{1,1}</shortName> <name>{0,1}</name> <description>{0,1}</description> <idLowerBound>{0,1}</idLowerBound> <idUpperBound>{0,1}</idUpperBound> <constantValue>{0,1}</constantValue> <iteratorContainer>{0,1}</iteratorContainer> <role>{0,1}</role> <mathType>{0,1}</mathType> <unit>{0,1}</unit> <formula>{0,1}</formula> <source>{0,1}</source> <group>{0,unbounded}</group> <modelSpec>{0,1}</modelSpec> </pre>

	<code></entitySpec></code>
Source	<code><xs:element name="entitySpec" type="m:entitySpec" maxOccurs="unbounded" /></code>

Element `m:entitySpec` / `id`

Namespace	No namespace
Diagram	
Type	xs:int
Properties	content: simple
Source	<code><xs:element name="id" type="xs:int" /></code>

Element `m:entitySpec` / `shortName`

Namespace	No namespace				
Diagram					
Type	m:shortName				
Properties	content: simple				
Facets	<table> <tr> <td>minLength</td><td>1</td></tr> <tr> <td>maxLength</td><td>8</td></tr> </table>	minLength	1	maxLength	8
minLength	1				
maxLength	8				
Source	<code><xs:element name="shortName" type="m:shortName" /></code>				

Element `m:entitySpec` / `name`

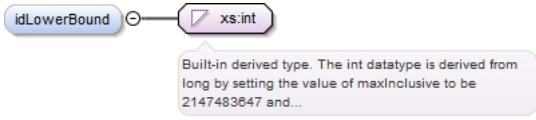
Namespace	No namespace						
Diagram							
Type	m:name						
Properties	<table> <tr> <td>content:</td><td>simple</td></tr> <tr> <td>minOccurs:</td><td>0</td></tr> <tr> <td>maxOccurs:</td><td>1</td></tr> </table>	content:	simple	minOccurs:	0	maxOccurs:	1
content:	simple						
minOccurs:	0						
maxOccurs:	1						
Facets	<table> <tr> <td>minLength</td><td>1</td></tr> <tr> <td>maxLength</td><td>32</td></tr> </table>	minLength	1	maxLength	32		
minLength	1						
maxLength	32						
Source	<code><xs:element name="name" type="m:name" minOccurs="0" maxOccurs="1" /></code>						

Element `m:entitySpec` / `description`

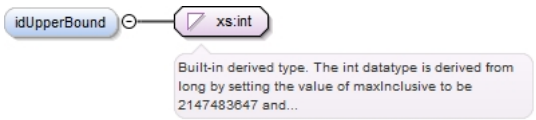
Namespace	No namespace						
Diagram							
Type	m:description						
Properties	<table> <tr> <td>content:</td><td>simple</td></tr> <tr> <td>minOccurs:</td><td>0</td></tr> <tr> <td>maxOccurs:</td><td>1</td></tr> </table>	content:	simple	minOccurs:	0	maxOccurs:	1
content:	simple						
minOccurs:	0						
maxOccurs:	1						
Facets	<table> <tr> <td>minLength</td><td>1</td></tr> <tr> <td>maxLength</td><td>512</td></tr> </table>	minLength	1	maxLength	512		
minLength	1						
maxLength	512						
Source	<code><xs:element name="description" type="m:description" minOccurs="0" maxOccurs="1" /></code>						

Element `m:entitySpec` / `idLowerBound`

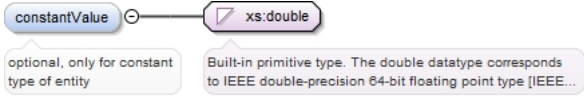
Namespace	No namespace
-----------	--------------

Diagram	 <p>Diagram showing the <code>idLowerBound</code> element (blue box) connected to the <code>xs:int</code> datatype (purple box). A callout box explains: "Built-in derived type. The int datatype is derived from long by setting the value of maxInclusive to be 2147483647 and..."</p>						
Type	<code>xs:int</code>						
Properties	<table> <tr> <td>content:</td><td>simple</td></tr> <tr> <td>minOccurs:</td><td>0</td></tr> <tr> <td>maxOccurs:</td><td>1</td></tr> </table>	content:	simple	minOccurs:	0	maxOccurs:	1
content:	simple						
minOccurs:	0						
maxOccurs:	1						
Source	<code><xs:element name="idLowerBound" type="xs:int" minOccurs="0" maxOccurs="1"/></code>						

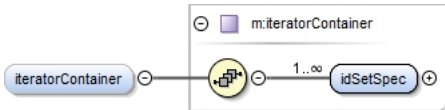
Element `m:entitySpec` / `idUpperBound`

Namespace	No namespace						
Diagram	 <p>Diagram showing the <code>idUpperBound</code> element (blue box) connected to the <code>xs:int</code> datatype (purple box). A callout box explains: "Built-in derived type. The int datatype is derived from long by setting the value of maxInclusive to be 2147483647 and..."</p>						
Type	<code>xs:int</code>						
Properties	<table> <tr> <td>content:</td><td>simple</td></tr> <tr> <td>minOccurs:</td><td>0</td></tr> <tr> <td>maxOccurs:</td><td>1</td></tr> </table>	content:	simple	minOccurs:	0	maxOccurs:	1
content:	simple						
minOccurs:	0						
maxOccurs:	1						
Source	<code><xs:element name="idUpperBound" type="xs:int" minOccurs="0" maxOccurs="1"/></code>						

Element `m:entitySpec` / `constantValue`

Namespace	No namespace						
Annotations	optional, only for constant type of entity						
Diagram	 <p>Diagram showing the <code>constantValue</code> element (blue box) connected to the <code>xs:double</code> datatype (purple box). Callouts explain: "optional, only for constant type of entity" and "Built-in primitive type. The double datatype corresponds to IEEE double-precision 64-bit floating point type [IEEE...]"</p>						
Type	<code>xs:double</code>						
Properties	<table> <tr> <td>content:</td><td>simple</td></tr> <tr> <td>minOccurs:</td><td>0</td></tr> <tr> <td>maxOccurs:</td><td>1</td></tr> </table>	content:	simple	minOccurs:	0	maxOccurs:	1
content:	simple						
minOccurs:	0						
maxOccurs:	1						
Source	<pre> <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> </pre>						

Element `m:entitySpec` / `iteratorContainer`


Namespace	No namespace						
Diagram	 <p>Diagram showing the <code>iteratorContainer</code> element (blue box) connected to a complex type (yellow box) which is connected to the <code>idSetSpec</code> element (blue box). A callout box explains: "m.iteratorContainer". The <code>idSetSpec</code> element has a cardinality of 1..∞.</p>						
Type	<code>m:iteratorContainer</code>						
Properties	<table> <tr> <td>content:</td><td>complex</td></tr> <tr> <td>minOccurs:</td><td>0</td></tr> <tr> <td>maxOccurs:</td><td>1</td></tr> </table>	content:	complex	minOccurs:	0	maxOccurs:	1
content:	complex						
minOccurs:	0						
maxOccurs:	1						
Model	<code>idSetSpec+</code>						
Children	<code>idSetSpec</code>						

Instance	<pre><iteratorContainer> <idSetSpec>{1,unbounded}</idSetSpec> </iteratorContainer></pre>
Source	<pre><xs:element name="iteratorContainer" type="m:iteratorContainer" minOccurs="0" maxOccurs="1"/></pre>

Element `m:entitySpec` / `role`

Namespace	No namespace		
Diagram			
Type	m:entityRole		
Properties	content:	simple	
	minOccurs:	0	
	maxOccurs:	1	
Facets	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
Source	<pre><xs:element name="role" type="m:entityRole" minOccurs="0" maxOccurs="1"/></pre>		

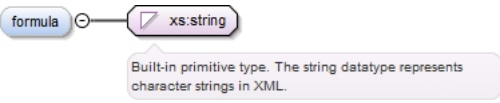
Element `m:entitySpec` / `mathType`

Namespace	No namespace		
Diagram			
Type	m:mathType		
Properties	content:	simple	
	minOccurs:	0	
	maxOccurs:	1	
Facets	enumeration	INTEGER	
	enumeration	BINARY	
	enumeration	REAL	
Source	<xs:element name="mathType" type="m:mathType" minOccurs="0" maxOccurs="1"/>		

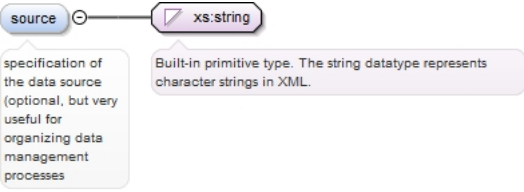
Element `m:entitySpec` / `unit`

Namespace	No namespace		
Diagram	<p>unit</p> <p>xs:string</p> <p>Built-in primitive type. The string datatype represents character strings in XML.</p>		
Type	xs:string		
Properties	content:	simple	
	minOccurs:	0	
	maxOccurs:	1	
Source	<xs:element name="unit" type="xs:string" minOccurs="0" maxOccurs="1"/>		

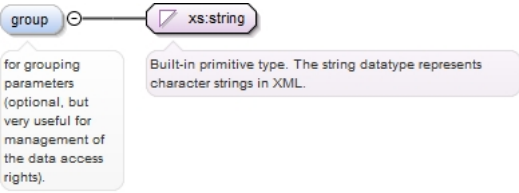
Element m:entitySpec / formula

Namespace	No namespace						
Diagram							
Type	xs:string						
Properties	<table> <tr> <td>content:</td><td>simple</td></tr> <tr> <td>minOccurs:</td><td>0</td></tr> <tr> <td>maxOccurs:</td><td>1</td></tr> </table>	content:	simple	minOccurs:	0	maxOccurs:	1
content:	simple						
minOccurs:	0						
maxOccurs:	1						
Source	<code><xs:element name="formula" type="xs:string" minOccurs="0" maxOccurs="1"/></code>						

Element m:entitySpec / source

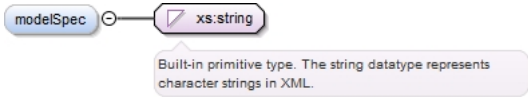
Namespace	No namespace						
Annotations	specification of the data source (optional, but very useful for organizing data management processes)						
Diagram							
Type	xs:string						
Properties	<table> <tr> <td>content:</td><td>simple</td></tr> <tr> <td>minOccurs:</td><td>0</td></tr> <tr> <td>maxOccurs:</td><td>1</td></tr> </table>	content:	simple	minOccurs:	0	maxOccurs:	1
content:	simple						
minOccurs:	0						
maxOccurs:	1						
Source	<pre> <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> </pre>						

Element m:entitySpec / group

Namespace	No namespace						
Annotations	for grouping parameters (optional, but very useful for management of the data access rights).						
Diagram							
Type	xs:string						
Properties	<table> <tr> <td>content:</td><td>simple</td></tr> <tr> <td>minOccurs:</td><td>0</td></tr> <tr> <td>maxOccurs:</td><td>unbounded</td></tr> </table>	content:	simple	minOccurs:	0	maxOccurs:	unbounded
content:	simple						
minOccurs:	0						
maxOccurs:	unbounded						
Source	<pre> <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> </pre>						

Element m:entitySpec / modelSpec

Namespace	No namespace
-----------	--------------

Diagram							
Type	xs:string						
Properties	<table><tr><td>content:</td><td>simple</td></tr><tr><td>minOccurs:</td><td>0</td></tr><tr><td>maxOccurs:</td><td>1</td></tr></table>	content:	simple	minOccurs:	0	maxOccurs:	1
content:	simple						
minOccurs:	0						
maxOccurs:	1						
Source	<pre><xs:element name="modelSpec" type="xs:string" minOccurs="0" maxOccurs="1" /></pre>						