

Schema documentation for data.xsd

september 11, 2012

Table of Contents

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

Resource hierarchy:

Legend:  Import,  Include,  Redefine,  Cycle detected

data.xsd

  sms.xsd

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

Schema(s)

Main 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	Element 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="xs:string" /> <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" type="xs:double"/> </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 index 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 i:dataSet/entityValues
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 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="xs:double"/> </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

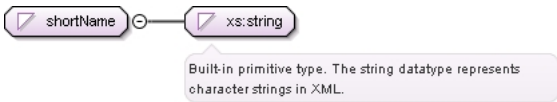
Simple Type(s)

Simple Type m:description

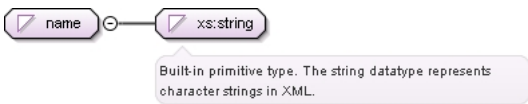
Namespace	http://www.ime.iiasa.ac.at/model/spec
Diagram	
Type	restriction of xs:string

Facets	minLength	1
	maxLength	512
Used by	Elements	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>	

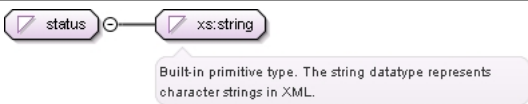
Simple Type m:shortName

Namespace	http://www.ime.iiasa.ac.at/model/spec	
Diagram		
Type	restriction of xs:string	
Facets	minLength	1
	maxLength	8
Used by	Elements	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	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:status

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

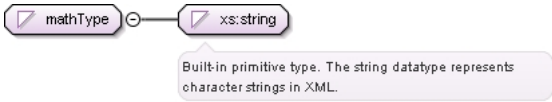
Namespace	http://www.ime.iiasa.ac.at/model/spec		
Diagram	<div><div>entityRole</div><div>xs:string</div><div>Built-in primitive type. The string datatype represents character strings in XML.</div></div>		
Type	restriction of xs:string		
Facets	enumeration	CONSTANT	
	enumeration	PARAMETER	
	enumeration	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:restriction> </xs:simpleType></pre>		

```

</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>

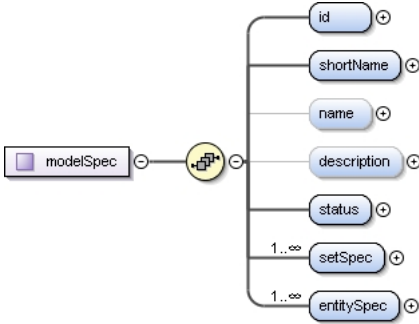
```

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>BINARY</td></tr> <tr> <td>enumeration</td><td>REAL</td></tr> </table>	enumeration	INTEGER	enumeration	BINARY	enumeration	REAL
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 <code>m:modelSpec/setSpec</code>
Model	<code>id</code> , <code>idParent</code> , <code>shortName</code> , <code>name</code> {0,1} , <code>description</code> {0,1} , <code>type</code> , <code>idx</code> {0,1} , <code>iteratorContainer</code> {0,1}
Children	<code>description</code> , <code>id</code> , <code>idParent</code> , <code>idx</code> , <code>iteratorContainer</code> , <code>name</code> , <code>shortName</code> , <code>type</code>
Source	<pre> <xs:complexType name="setSpec"> <xs:sequence> <xs:element name="id" type="xs:int"/> <!--sub set--> <xs:element name="idParent" 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="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	<code>http://www.ime.iiasa.ac.at/model/spec</code>
Diagram	
Used by	Elements <code>m:entitySpec/iteratorContainer</code> , <code>m:setSpec/iteratorContainer</code>
Model	<code>idSetSpec</code> +
Children	<code>idSetSpec</code>
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	<code>http://www.ime.iiasa.ac.at/model/spec</code>
-----------	--

Diagram	
Used by	Element <code>m:modelSpec/entitySpec</code>
Model	<code>id</code> , <code>shortName</code> , <code>name{0,1}</code> , <code>description{0,1}</code> , <code>idLowerBound{0,1}</code> , <code>idUpperBound{0,1}</code> , <code>constantValue{0,1}</code> , <code>iteratorContainer{0,1}</code> , <code>role{0,1}</code> , <code>mathType{0,1}</code> , <code>unit{0,1}</code> , <code>formula{0,1}</code> , <code>source{0,1}</code> , <code>group*</code>
Children	<code>constantValue</code> , <code>description</code> , <code>formula</code> , <code>group</code> , <code>id</code> , <code>idLowerBound</code> , <code>idUpperBound</code> , <code>iteratorContainer</code> , <code>mathType</code> , <code>name</code> , <code>role</code> , <code>shortName</code> , <code>source</code> , <code>unit</code>
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="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:sequence> </xs:complexType> </pre>