Schema documentation for config.xsd

december 22, 2018

Table of Contents

Namespace: ""

Schema(s)

Main schema `config.xsd`

Namespace	No namespace	
Properties	attribute form default:	`unqualified`
	element form default:	`unqualified`
	version:	`1.0`

Element(s)

Element `config`

Namespace	No namespace			
Type	config			
Properties	content:	`complex`		
Model	database{0,1} , files , data-sets{0,1}			
Children	data-sets, database, files			

Instance	`<config>`
	`  <database name="">{0,1}</database>`
	`  <files>{1,1}</files>`
	`  <data-sets>{0,1}</data-sets>`
	`</config>`			

Source	`<xs:element name="config" type="config">`
	`  <xs:key name="dataFileIdKey">`
	`    <xs:selector xpath="files/file"/>`
	`    <xs:field xpath="@id"/>`
	`  </xs:key>`
	`  <xs:keyref refer="dataFileIdKey" name="dataFileIdKeyRef">`
	`    <xs:selector xpath="data-sets/dataset/source"/>`
	`    <xs:field xpath="@file"/>`
	`  </xs:keyref>`
	`</xs:element>`			

Element `config` / `database`

Namespace	No namespace			
Annotations	MySql (or compatible) database configuration. Default database configuration is: name = magemediation, server.host = localhost, server.port = 3306, server.user = root, server.password = root			
Type	database			
Properties	content:	`complex`		
	minOccurs:	`0`		
Model	server			
Children	server			

Instance	`<database name="">`
	`  <server host="" password="" port="" user="">{1,1}</server>`
	`</database>`			

Attributes	**QName**	**Type**	**Use**	
	name	xs:string	required	

QName	Type	Use	
	MySql database name		

Source	`<xs:element name="database" type="database" minOccurs="0">`
	  `<xs:annotation>`
	    `<xs:documentation>`MySql (or compatible) database configuration. Default database configuration is: name = magemediation, server.host = localhost, server.port = 3306, server.user = root, server.password = root`</xs:documentation>`
	  `</xs:annotation>`
	`</xs:element>`		

Element `database / server`

Namespace	No namespace			
Type	databaseServer			
Properties	content: complex			
Attributes	**QName**	**Type**	**Use**	
	host	xs:string	required	
	password	xs:string	required	
	port	xs:int	optional	
	user	xs:string	required	
Source	`<xs:element name="server" type="databaseServer"/>`			

Element `config / files`

Namespace	No namespace
Annotations	This section configures list of files that contain data to process. Allowable file types: xls, xlsx, csv.
Properties	content: complex
Model	file*
Children	file

Instance	`<files>`
	  `<file id="" path="">`{0,unbounded}`</file>`
	`</files>`

Source	`<xs:element name="files">`
	  `<xs:annotation>`
	    `<xs:documentation>`This section configures list of files that contain data to process. Allowable file types: xls, xlsx, csv.`</xs:documentation>`
	  `</xs:annotation>`
	  `<xs:complexType>`
	    `<xs:sequence>`
	      `<xs:element name="file" type="dataFile" nillable="true" minOccurs="0" maxOccurs="unbounded">`
	        `<xs:annotation>`
	          `<xs:documentation>`Data file path. ID should be used as reference in dataset sources.`</xs:documentation>`
	        `</xs:annotation>`
	      `</xs:element>`
	    `</xs:sequence>`
	  `</xs:complexType>`
	`</xs:element>`

Element `config / files / file`

Namespace	No namespace			
Annotations	Data file path. ID should be used as reference in dataset sources.			
Type	dataFile			
Properties	content: complex			
	minOccurs: 0			
	maxOccurs: unbounded			
	nillable: true			
Attributes	**QName**	**Type**	**Use**	
	id	xs:token	required	
		Data source file id, to use in next configuration		

QName	Type	Use	
path	xs:string	required	
	Data source file path		

Source	`<xs:element name="file" type="dataFile" nillable="true" minOccurs="0" maxOccurs="unbounded">`
	`  <xs:annotation>`
	`    <xs:documentation>Data file path. ID should be used as reference in dataset sources.</`
	`xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

Element `config / data-sets`

Namespace	No namespace
Properties	content: complex minOccurs: 0
Model	dataset*
Children	dataset

Instance	`<data-sets>`
	`  <dataset headers-row="" max-absent-age="" name="" rows-to-skip="" target="">{0,unbounded}</`
	`dataset>`
	`</data-sets>`

Source	`<xs:element name="data-sets" minOccurs="0">`
	`  <xs:complexType>`
	`    <xs:sequence>`
	`      <xs:element name="dataset" type="data-set" nillable="true" minOccurs="0"`
	` maxOccurs="unbounded">`
	`        <xs:annotation>`
	`          <xs:documentation>Describe source data that should be loaded to mediation database.</`
	`xs:documentation>`
	`        </xs:annotation>`
	`      </xs:element>`
	`    </xs:sequence>`
	`  </xs:complexType>`
	`  <xs:unique name="datasetNameUnique">`
	`    <xs:selector xpath="dataset"/>`
	`    <xs:field xpath="@name"/>`
	`  </xs:unique>`
	`</xs:element>`

Element `config / data-sets / dataset`

Namespace	No namespace
Annotations	Describe source data that should be loaded to mediation database.
Type	data-set
Properties	content: complex minOccurs: 0 maxOccurs: unbounded nillable: true
Model	source+ , field-sets
Children	field-sets, source

Instance	`<dataset headers-row="" max-absent-age="" name="" rows-to-skip="" target="">`
	`  <source empty-row="ignore" file="" sheet="">{1,unbounded}</source>`
	`  <field-sets>{1,1}</field-sets>`
	`</dataset>`

Attributes	QName	Type	Use	
	headers-row	xs:nonNegativeInteger	optional	
		Zero base row number that contains headers. If it is not defined - the data source does not contian headers.		
	max-absent-age	xs:int	optional	
		Specify number of days to store objects in mediation database that does not exist in data soutrce.		
	name	xs:string	required	
		Internal dataset id, should be uniqe		

QName	Type	Use	
rows-to-skip	xs:int	optional	
	Number of rows to skip before real data		
target	datasetTarget	required	
	Destination mediation table		

Asserts	Test		XPath default namespace
	if (exists(@headers-row)) then true() else count(field-sets/*/ field/@column) = count(field-sets/*/field)		

Source	`<xs:element name="dataset" type="data-set" nillable="true" minOccurs="0" maxOccurs="unbounded">`
	  `<xs:annotation>`
	    `<xs:documentation>Describe source data that should be loaded to mediation database.</`
	`xs:documentation>`
	  `</xs:annotation>`
	`</xs:element>`

Element `data-set / source`

Namespace	No namespace				
Type	dataSource				
Properties	content:	complex			
	minOccurs:	1			
	maxOccurs:	unbounded			
Attributes	**QName**	**Type**	**Default**	**Use**	
	empty-row	emptyRowAction	ignore	optional	
	file	xs:string		required	
		Reference to file in files section			
	sheet	xs:string		required	
		Name of sheet contains data. For CSV files only one sheet supported - default			
Source	`<xs:element name="source" type="dataSource" minOccurs="1" maxOccurs="unbounded"/>`				

Element `data-set / field-sets`

Namespace	No namespace	
Properties	content:	complex
	minOccurs:	1
	maxOccurs:	1
Model	main , additional*	
Children	additional, main	

Instance	`<field-sets>`
	  `<main name="">`{1,1}`</main>`
	  `<additional name="">`{0,unbounded}`</additional>`
	`</field-sets>`	

Source	`<xs:element name="field-sets" minOccurs="1" maxOccurs="1">`
	  `<xs:complexType>`
	    `<xs:sequence>`
	      `<xs:element name="main" type="fieldSet" minOccurs="1" maxOccurs="1">`
	        `<xs:unique name="mainFieldColumnUnique">`
	          `<xs:selector xpath="field"/>`
	          `<xs:field xpath="@column"/>`
	        `</xs:unique>`
	        `<xs:key name="mainParentFieldKey">`
	          `<xs:selector xpath="field"/>`
	          `<xs:field xpath="@name"/>`
	        `</xs:key>`
	        `<xs:keyref name="mainParentFieldKeyRef" refer="mainParentFieldKey">`
	          `<xs:selector xpath="field"/>`
	          `<xs:field xpath="@parent"/>`
	        `</xs:keyref>`
	      `</xs:element>`
	      `<xs:element name="additional" type="fieldSet" minOccurs="0" maxOccurs="unbounded">`
	        `<xs:unique name="commonF">`

```
                    <xs:selector xpath="field"/>
                    <xs:field xpath="@column"/>
                </xs:unique>
                <xs:key name="commonParentFieldKey">
                    <xs:selector xpath="field"/>
                    <xs:field xpath="@name"/>
                </xs:key>
                <xs:keyref name="commonParentFieldKeyRef" refer="commonParentFieldKey">
                    <xs:selector xpath="field"/>
                    <xs:field xpath="@parent"/>
                </xs:keyref>
            </xs:element>
        </xs:sequence>
    </xs:complexType>
    <xs:key name="fieldNameKey">
        <xs:selector xpath="*/field"/>
        <xs:field xpath="@name"/>
    </xs:key>
</xs:element>
```

Element `data-set / field-sets / main`

Namespace	No namespace			
Type	fieldSet			
Properties	content:	complex		
	minOccurs:	1		
	maxOccurs:	1		
	mixed:	false		
Model	field+			
Children	field			

Instance	`<main name="">`
	`  <field column="" key-field="" locale="" name="" nested="" optional="false" parent="" regex="" type="string">`{1,unbounded}`</field>`
	`</main>`			

Attributes	**QName**	**Type**	**Use**	
	name	xs:string	required	
		Attribute name that should fit to name in data source. This name will be as attribute name in mediation.		
Asserts	**Test**		**XPath default namespace**	
	if (name(.) = 'main') then exists(field/@key-field) and field/@key-field = true() else true()			
	One field in main field set should have key-field attribute equals to true.			

Source	`<xs:element name="main" type="fieldSet" minOccurs="1" maxOccurs="1">`
	`  <xs:unique name="mainFieldColumnUnique">`
	`    <xs:selector xpath="field"/>`
	`    <xs:field xpath="@column"/>`
	`  </xs:unique>`
	`  <xs:key name="mainParentFieldKey">`
	`    <xs:selector xpath="field"/>`
	`    <xs:field xpath="@name"/>`
	`  </xs:key>`
	`  <xs:keyref name="mainParentFieldKeyRef" refer="mainParentFieldKey">`
	`    <xs:selector xpath="field"/>`
	`    <xs:field xpath="@parent"/>`
	`  </xs:keyref>`
	`</xs:element>`			

Element `fieldSet / field`

Namespace	No namespace
Type	field
Properties	content: complex
	maxOccurs: unbounded
Model	list-format{0,1} , date-format{0,1} , datetime-format{0,1} , attr-list-format{0,1}
Children	attr-list-format, date-format, datetime-format, list-format

Instance	<pre><field column="" key-
	field="" locale="" name="" nested="" optional="false" parent="" regex="" type="string">
	  <list-format delimeter="" quote="">{0,1}</list-format>
	  <date-format format="">{0,1}</date-format>
	  <datetime-format format="">{0,1}</datetime-format>
	  <attr-list-format attr-delimeter="" attr-quote="" name-value-delimiter="" value-quote="">{0,1}</
	attr-list-format>
	</field></pre>			

Attributes	**QName**	**Type**	**Default**	**Use**	
	column	xs:nonNegativeInteger		optional	
	key-field	xs:boolean		optional	
	locale	xs:string		optional	
		Language and country code like ru_RU, en_US			
	name	xs:string		required	
	nested	xs:boolean		optional	
	optional	xs:boolean	false	optional	
	parent	xs:string		optional	
	regex	xs:string		optional	
	type	fieldType	string	optional	
Asserts	**Test**		**XPath default namespace**		
	if (@type = 'bool-list' or @type = 'int-list' or @type = 'double-list' or @type = 'string-list') then exists(list-format) else if (@type = 'date-list') then (exists(list-format) and exists(date-format)) else if (@type = 'datetime-list') then (exists(list-format) and exists(datetime-format)) else if (@type = 'date') then exists(date-format) else if (@type = 'date') then exists(datetime-format) else if (@type = 'attribute-list') then exists(attr-list-format) else true()				
	if (@nested = true()) then exists(@parent) else true()				
	if (@nested = true()) then not(@key-field) else true()				
Source	`<xs:element name="field" type="field" maxOccurs="unbounded"/>`				

Element `field / list-format`

Namespace	No namespace		
Type	listDefinition		
Properties	content:	complex	
	minOccurs:	0	
Attributes	**QName**	**Type**	**Use**	
	delimeter	xs:string	required	
	quote	xs:string	required	
Source	`<xs:element name="list-format" type="listDefinition" minOccurs="0"/>`			

Element `field / date-format`

Namespace	No namespace

Annotations	<pre>Symbol Meaning Presentation Examples
	------ ------- ------------ -------
	 G era text AD
	 C century of era (>=0) number 20
	 Y year of era (>=0) year 1996
	
	 x weekyear year 1996
	 w week of weekyear number 27
	 e day of week number 2
	 E day of week text Tuesday; Tue
	
	 y year year 1996
	 D day of year number 189
	 M month of year month July; Jul; 07
	 d day of month number 10
	
	 a halfday of day text PM</pre>

```
        K       hour of halfday (0~11)      number      0
        h       clockhour of halfday (1~12) number      12

        H       hour of day (0~23)          number      0
        k       clockhour of day (1~24)     number      24
        m       minute of hour              number      30
        s       second of minute            number      55
        S       fraction of second          number      978

        z       time zone                   text        Pacific Standard Time; PST
        Z       time zone offset/id         zone        -0800; -08:00; America/Los_Angeles

        '       escape for text             delimiter
        ''      single quote                literal     '
```

Type	dateDefinition		
Properties	content: complex		
	minOccurs: 0		
Attributes	**QName**	**Type**	**Use**
	format	xs:string	optional

Source	`<xs:element name="date-format" type="dateDefinition" minOccurs="0">`
	  `<xs:annotation>`
	    `<xs:documentation>`Symbol Meaning Presentation Examples ------ ------- ------------ ------- G era text AD C century of era (>=0) number 20 Y year of era (>=0) year 1996 x weekyear year 1996 w week of weekyear number 27 e day of week number 2 E day of week text Tuesday; Tue y year year 1996 D day of year number 189 M month of year month July; Jul; 07 d day of month number 10 a halfday of day text PM K hour of halfday (0~11) number 0 h clockhour of halfday (1~12) number 12 H hour of day (0~23) number 0 k clockhour of day (1~24) number 24 m minute of hour number 30 s second of minute number 55 S fraction of second number 978 z time zone text Pacific Standard Time; PST Z time zone offset/id zone -0800; -08:00; America/Los_Angeles ' escape for text delimiter '' single quote literal '`</xs:documentation>`
	  `</xs:annotation>`
	`</xs:element>`		

Element **field / datetime-format**

Namespace	No namespace		
Type	datetimeDefinition		
Properties	content: complex		
	minOccurs: 0		
Attributes	**QName**	**Type**	**Use**
	format	xs:string	optional
Source	`<xs:element name="datetime-format" type="datetimeDefinition" minOccurs="0"/>`		

Element **field / attr-list-format**

Namespace	No namespace
| Annotations | ```
Symbol  Meaning                     Presentation  Examples
------  -------                     ------------  -------
G       era                         text          AD
C       century of era (>=0)        number        20
Y       year of era (>=0)           year          1996

x       weekyear                    year          1996
w       week of weekyear            number        27
e       day of week                 number        2
E       day of week                 text          Tuesday; Tue

y       year                        year          1996
D       day of year                 number        189
M       month of year               month         July; Jul; 07
d       day of month                number        10

a       halfday of day              text          PM
K       hour of halfday (0~11)      number        0
h       clockhour of halfday (1~12) number        12

H       hour of day (0~23)          number        0
k       clockhour of day (1~24)     number        24
m       minute of hour              number        30
s       second of minute            number        55
S       fraction of second          number        978
``` |

| | ```
z       time zone               text       Pacific Standard Time; PST
Z       time zone offset/id      zone       -0800; -08:00; America/Los_Angeles

'       escape for text          delimiter
''      single quote             literal    '
``` |
Type	attrListDefinition
Properties	content: complex minOccurs: 0
Attributes	

QName	Type	Use	
attr-delimeter	xs:string	required	
attr-quote	xs:string	required	
name-value-delimiter	xs:string	optional	
value-quote	xs:string	optional	

| Source | ```xml
<xs:element minOccurs="0" name="attr-list-format" type="attrListDefinition">
  <xs:annotation>
    <xs:documentation>Symbol Meaning Presentation Examples ------ ------- ----------- ------- G
 era text AD C century of era (>=0) number 20 Y year of era (>=0) year 1996 x weekyear year 1996 w
 week of weekyear number 27 e day of week number 2 E day of week text Tuesday; Tue y year year 1996
 D day of year number 189 M month of year month July; Jul; 07 d day of month number 10 a halfday
 of day text PM K hour of halfday (0~11) number 0 h clockhour of halfday (1~12) number 12 H hour
 of day (0~23) number 0 k clockhour of day (1~24) number 24 m minute of hour number 30 s second of
 minute number 55 S fraction of second number 978 z time zone text Pacific Standard Time; PST Z time
 zone offset/id zone -0800; -08:00; America/Los_Angeles ' escape for text delimiter '' single quote
 literal '</xs:documentation>
  </xs:annotation>
</xs:element>
``` |

Element `data-set / field-sets / additional`

Namespace	No namespace
Type	fieldSet
Properties	content: complex minOccurs: 0 maxOccurs: unbounded mixed: false
Model	field+
Children	field
| Instance | ```xml
<additional name="">
  <field column="" key-
field="" locale="" name="" nested="" optional="false" parent="" regex="" type="string">{1,unbounded}</
field>
</additional>
``` |
| Attributes | |

QName	Type	Use	
name	xs:string	required	
	Attribute name that should fit to name in data source. This name will be as attribute name in mediation.		

Asserts	

Test	XPath default namespace
if (name(.) = 'main') then exists(field/@key-field) and field/@key-field = true() else true()	
One field in main field set should have key-field attribute equals to true.	

| Source | ```xml
<xs:element name="additional" type="fieldSet" minOccurs="0" maxOccurs="unbounded">
  <xs:unique name="commonF">
    <xs:selector xpath="field"/>
    <xs:field xpath="@column"/>
  </xs:unique>
  <xs:key name="commonParentFieldKey">
    <xs:selector xpath="field"/>
    <xs:field xpath="@name"/>
  </xs:key>
  <xs:keyref name="commonParentFieldKeyRef" refer="commonParentFieldKey">
    <xs:selector xpath="field"/>
    <xs:field xpath="@parent"/>
  </xs:keyref>
``` |

```
                                                               </xs:element>
```

Complex Type(s)

Complex Type `config`

Namespace	No namespace
Used by	Element config
Model	database{0,1} , files , data-sets{0,1}
Children	data-sets, database, files

Source	<pre><xs:complexType name="config">
	  <xs:sequence>
	    <xs:element name="database" type="database" minOccurs="0">
	      <xs:annotation>
	        <xs:documentation>MySql (or compatible) database configuration. Default database
	configuration is: name = magemediation, server.host = localhost, server.port = 3306, server.user =
	root, server.password = root</xs:documentation>
	      </xs:annotation>
	    </xs:element>
	    <xs:element name="files">
	      <xs:annotation>
	        <xs:documentation>This section configures list of files that contain data to process.
	Allowable file types: xls, xlsx, csv.</xs:documentation>
	      </xs:annotation>
	      <xs:complexType>
	        <xs:sequence>
	          <xs:element name="file" type="dataFile" nillable="true" minOccurs="0"
	 maxOccurs="unbounded">
	            <xs:annotation>
	              <xs:documentation>Data file path. ID should be used as reference in dataset sources.</
	xs:documentation>
	            </xs:annotation>
	          </xs:element>
	        </xs:sequence>
	      </xs:complexType>
	    </xs:element>
	    <xs:element name="data-sets" minOccurs="0">
	      <xs:complexType>
	        <xs:sequence>
	          <xs:element name="dataset" type="data-set" nillable="true" minOccurs="0"
	 maxOccurs="unbounded">
	            <xs:annotation>
	              <xs:documentation>Describe source data that should be loaded to mediation database.</
	xs:documentation>
	            </xs:annotation>
	          </xs:element>
	        </xs:sequence>
	      </xs:complexType>
	      <xs:unique name="datasetNameUnique">
	        <xs:selector xpath="dataset"/>
	        <xs:field xpath="@name"/>
	      </xs:unique>
	    </xs:element>
	  </xs:sequence>
	</xs:complexType></pre>

Complex Type `database`

Namespace	No namespace		
Used by	Element config/database		
Model	server		
Children	server		
Attributes	**QName**	**Type**	**Use**
	name	xs:string	required
		MySql database name	

Source	<pre><xs:complexType name="database">
	  <xs:sequence>
	    <xs:element name="server" type="databaseServer"/>
	  </xs:sequence>
	  <xs:attribute name="name" type="xs:string" use="required">
	    <xs:annotation>
	      <xs:documentation>MySql database name</xs:documentation>
	    </xs:annotation></pre>		

```
    </xs:attribute>
  </xs:complexType>
```

Complex Type `databaseServer`

Namespace	No namespace			
Used by	Element	database/server		
Attributes	**QName**	**Type**	**Use**	
	host	xs:string	required	
	password	xs:string	required	
	port	xs:int	optional	
	user	xs:string	required	

Source	`<xs:complexType name="databaseServer">`
	`  <xs:attribute name="host" type="xs:string" use="required"/>`
	`  <xs:attribute name="port" type="xs:int"/>`
	`  <xs:attribute name="user" type="xs:string" use="required"/>`
	`  <xs:attribute name="password" type="xs:string" use="required"/>`
	`</xs:complexType>`			

Complex Type `dataFile`

Namespace	No namespace			
Used by	Element	config/files/file		
Attributes	**QName**	**Type**	**Use**	
	id	xs:token	required	
		Data source file id, to use in next configuration		
	path	xs:string	required	
		Data source file path		

Source	`<xs:complexType name="dataFile">`
	`  <xs:attribute name="id" type="xs:token" use="required">`
	`    <xs:annotation>`
	`      <xs:documentation>Data source file id, to use in next configuration</xs:documentation>`
	`    </xs:annotation>`
	`  </xs:attribute>`
	`  <xs:attribute name="path" type="xs:string" use="required">`
	`    <xs:annotation>`
	`      <xs:documentation>Data source file path</xs:documentation>`
	`    </xs:annotation>`
	`  </xs:attribute>`
	`</xs:complexType>`			

Complex Type `data-set`

Namespace	No namespace			
Used by	Element	config/data-sets/dataset		
Model	source+ , field-sets			
Children	field-sets, source			
Attributes	**QName**	**Type**	**Use**	
	headers-row	xs:nonNegativeInteger	optional	
		Zero base row number that contains headers. If it is not defined - the data source does not contian headers.		
	max-absent-age	xs:int	optional	
		Specify number of days to store objects in mediation database that does not exist in data soutrce.		
	name	xs:string	required	
		Internal dataset id, should be uniqe		
	rows-to-skip	xs:int	optional	
		Number of rows to skip before real data		
	target	datasetTarget	required	
		Destination mediation table		

Asserts	Test	XPath default namespace
	if (exists(@headers-row)) then true() else count(field-sets/*/field/@column) = count(field-sets/*/field)	
Source	<pre><xs:complexType name="data-set">
<xs:sequence>
<xs:element name="source" type="dataSource" minOccurs="1" maxOccurs="unbounded"/>
<xs:element name="field-sets" minOccurs="1" maxOccurs="1">
<xs:complexType>
<xs:sequence>
<xs:element name="main" type="fieldSet" minOccurs="1" maxOccurs="1">
<xs:unique name="mainFieldColumnUnique">
<xs:selector xpath="field"/>
<xs:field xpath="@column"/>
</xs:unique>
<xs:key name="mainParentFieldKey">
<xs:selector xpath="field"/>
<xs:field xpath="@name"/>
</xs:key>
<xs:keyref name="mainParentFieldKeyRef" refer="mainParentFieldKey">
<xs:selector xpath="field"/>
<xs:field xpath="@parent"/>
</xs:keyref>
</xs:element>
<xs:element name="additional" type="fieldSet" minOccurs="0" maxOccurs="unbounded">
<xs:unique name="commonF">
<xs:selector xpath="field"/>
<xs:field xpath="@column"/>
</xs:unique>
<xs:key name="commonParentFieldKey">
<xs:selector xpath="field"/>
<xs:field xpath="@name"/>
</xs:key>
<xs:keyref name="commonParentFieldKeyRef" refer="commonParentFieldKey">
<xs:selector xpath="field"/>
<xs:field xpath="@parent"/>
</xs:keyref>
</xs:element>
</xs:sequence>
</xs:complexType>
<xs:key name="fieldNameKey">
<xs:selector xpath="*/field"/>
<xs:field xpath="@name"/>
</xs:key>
</xs:element>
</xs:sequence>
<xs:attribute name="name" type="xs:string" use="required">
<xs:annotation>
<xs:documentation>Internal dataset id, should be uniqe</xs:documentation>
</xs:annotation>
</xs:attribute>
<xs:attribute name="headers-row" type="xs:nonNegativeInteger">
<xs:annotation>
<xs:documentation>Zero base row number that contains headers. If it is not defined - the data source does not contian headers.</xs:documentation>
</xs:annotation>
</xs:attribute>
<xs:attribute name="rows-to-skip" type="xs:int">
<xs:annotation>
<xs:documentation>Number of rows to skip before real data</xs:documentation>
</xs:annotation>
</xs:attribute>
<xs:attribute name="max-absent-age" type="xs:int">
<xs:annotation>
<xs:documentation>Specify number of days to store objects in mediation database that does not exist in data soutrce.</xs:documentation>
</xs:annotation>
</xs:attribute>
<xs:attribute name="target" type="datasetTarget" use="required">
<xs:annotation>
<xs:documentation>Destination mediation table</xs:documentation>
</xs:annotation>
</xs:attribute>
<xs:assert test="   if (exists(@headers-row)) then   true()   else count(field-sets/*/field/@column) = count(field-sets/*/field)"/>
</xs:complexType></pre>

Complex Type `dataSource`

Namespace	No namespace

Used by	Element	data-set/source			
Attributes	**QName**	**Type**	**Default**	**Use**	
	empty-row	emptyRowAction	ignore	optional	
	file	xs:string		required	
		Reference to file in files section			
	sheet	xs:string		required	
		Name of sheet contains data. For CSV files only one sheet supported - default			

Source	`<xs:complexType name="dataSource">`
	`  <xs:attribute name="file" type="xs:string" use="required">`
	`    <xs:annotation>`
	`      <xs:documentation>Reference to file in files section</xs:documentation>`
	`    </xs:annotation>`
	`  </xs:attribute>`
	`  <xs:attribute name="sheet" type="xs:string" use="required">`
	`    <xs:annotation>`
	`      <xs:documentation>Name of sheet contains data. For CSV files only one sheet supported -`
	` default</xs:documentation>`
	`    </xs:annotation>`
	`  </xs:attribute>`
	`  <xs:attribute name="empty-row" default="ignore" type="emptyRowAction"/>`
	`</xs:complexType>`				

Complex Type `fieldSet`

Namespace	No namespace		
Properties	mixed:	false	
Used by	Elements	data-set/field-sets/additional, data-set/field-sets/main	
Model	field+		
Children	field		
Attributes	**QName**	**Type**	**Use**	
	name	xs:string	required	
		Attribute name that should fit to name in data source. This name will be as attribute name in mediation.		
Asserts	**Test**		**XPath default namespace**
	if (name(.) = 'main') then exists(field/@key-field) and field/@key-field = true() else true()		
	One field in main field set should have key-field attribute equals to true.		

Source	`<xs:complexType name="fieldSet" mixed="false">`
	`  <xs:sequence>`
	`    <xs:element name="field" type="field" maxOccurs="unbounded"/>`
	`  </xs:sequence>`
	`  <xs:attribute name="name" type="xs:string" use="required">`
	`    <xs:annotation>`
	`      <xs:documentation>Attribute name that should fit to name in data source. This name will be as`
	` attribute name in mediation.</xs:documentation>`
	`    </xs:annotation>`
	`  </xs:attribute>`
	`  <xs:assert test="          if (name(.) = 'main') then           exists(field/@key-field) and field/`
	`@key-field = true()          else           true()">`
	`    <xs:annotation>`
	`      <xs:documentation>One field in main field set should have key-field attribute equals to`
	` true.</xs:documentation>`
	`    </xs:annotation>`
	`  </xs:assert>`
	`</xs:complexType>`		

Complex Type `field`

Namespace	No namespace	
Properties	final:	extension, restriction
Used by	Element	fieldSet/field
Model	list-format{0,1} , date-format{0,1} , datetime-format{0,1} , attr-list-format{0,1}	

Children	attr-list-format, date-format, datetime-format, list-format				
Attributes	**QName**	**Type**	**Default**	**Use**	
	column	xs:nonNegativeInteger		optional	
	key-field	xs:boolean		optional	
	locale	xs:string		optional	
		Language and country code like ru_RU, en_US			
	name	xs:string		required	
	nested	xs:boolean		optional	
	optional	xs:boolean	false	optional	
	parent	xs:string		optional	
	regex	xs:string		optional	
	type	fieldType	string	optional	

Asserts	**Test**	**XPath default namespace**
	if (@type = 'bool-list' or @type = 'int-list' or @type = 'double-list' or @type = 'string-list') then exists(list-format) else if (@type = 'date-list') then (exists(list-format) and exists(date-format)) else if (@type = 'datetime-list') then (exists(list-format) and exists(datetime-format)) else if (@type = 'date') then exists(date-format) else if (@type = 'date') then exists(datetime-format) else if (@type = 'attribute-list') then exists(attr-list-format) else true()	
	if (@nested = true()) then exists(@parent) else true()	
	if (@nested = true()) then not(@key-field) else true()	

Source	

```xml
<xs:complexType name="field" final="#all">
  <xs:sequence minOccurs="0">
    <xs:element name="list-format" type="listDefinition" minOccurs="0"/>
    <xs:element name="date-format" type="dateDefinition" minOccurs="0">
      <xs:annotation>
        <xs:documentation>Symbol Meaning Presentation Examples ------ ------- ------------ ------- G
 era text AD C century of era (>=0) number 20 Y year of era (>=0) year 1996 x weekyear year 1996 w
 week of weekyear number 27 e day of week number 2 E day of week text Tuesday; Tue y year year 1996
 D day of year number 189 M month of year month July; Jul; 07 d day of month number 10 a halfday
 of day text PM K hour of halfday (0~11) number 0 h clockhour of halfday (1~12) number 12 H hour
 of day (0~23) number 0 k clockhour of day (1~24) number 24 m minute of hour number 30 s second of
 minute number 55 S fraction of second number 978 z time zone text Pacific Standard Time; PST Z time
 zone offset/id zone -0800; -08:00; America/Los_Angeles ' escape for text delimiter '' single quote
 literal '</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="datetime-format" type="datetimeDefinition" minOccurs="0"/>
    <xs:element minOccurs="0" name="attr-list-format" type="attrListDefinition">
      <xs:annotation>
        <xs:documentation>Symbol Meaning Presentation Examples ------ ------- ------------ ------- G
 era text AD C century of era (>=0) number 20 Y year of era (>=0) year 1996 x weekyear year 1996 w
 week of weekyear number 27 e day of week number 2 E day of week text Tuesday; Tue y year year 1996
 D day of year number 189 M month of year month July; Jul; 07 d day of month number 10 a halfday
 of day text PM K hour of halfday (0~11) number 0 h clockhour of halfday (1~12) number 12 H hour
 of day (0~23) number 0 k clockhour of day (1~24) number 24 m minute of hour number 30 s second of
 minute number 55 S fraction of second number 978 z time zone text Pacific Standard Time; PST Z time
 zone offset/id zone -0800; -08:00; America/Los_Angeles ' escape for text delimiter '' single quote
 literal '</xs:documentation>
      </xs:annotation>
    </xs:element>
  </xs:sequence>
  <xs:attribute name="name" type="xs:string" use="required"/>
  <xs:attribute name="column" type="xs:nonNegativeInteger"/>
  <xs:attribute name="regex" type="xs:string"/>
  <xs:attribute name="type" type="fieldType" default="string"/>
  <xs:attribute name="key-field" type="xs:boolean"/>
  <xs:attribute name="nested" type="xs:boolean" use="optional"/>
  <xs:attribute name="parent" type="xs:string" use="optional"/>
  <xs:attribute default="false" name="optional" type="xs:boolean"/>
  <xs:attribute name="locale" type="xs:string">
    <xs:annotation>
      <xs:documentation>Language and country code like ru_RU, en_US</xs:documentation>
    </xs:annotation>
  </xs:attribute>
  <xs:assert test="          if (@type = 'bool-list' or @type = 'int-list' or @type = 'double-list'
 or @type = 'string-list') then          exists(list-format)          else          if (@type =
 'date-list') then          (exists(list-format) and exists(date-format))          else
```

```
                if (@type = 'datetime-list') then            (exists(list-format) and exists(datetime-
format))                else               if (@type = 'date') then               exists(date-format)
            else              if (@type = 'date') then              exists(datetime-
format)               else                 if (@type = 'attribute-list') then
  exists(attr-list-format)              else                 true()"/>
      <xs:assert test="         if (@nested = true()) then          exists(@parent)         else
        true()"/>
      <xs:assert test="          if (@nested = true()) then          not(@key-field)         else
        true()"/>
</xs:complexType>
```

Complex Type `listDefinition`

Namespace	No namespace			
Used by	Element	field/list-format		
Attributes	**QName**	**Type**	**Use**	
	delimeter	xs:string	required	
	quote	xs:string	required	

Source	`<xs:complexType name="listDefinition">`
	`  <xs:attribute name="delimeter" type="xs:string" use="required"/>`
	`  <xs:attribute name="quote" type="xs:string" use="required"/>`
	`</xs:complexType>`			

Complex Type `dateDefinition`

Namespace	No namespace			
Used by	Element	field/date-format		
Attributes	**QName**	**Type**	**Use**	
	format	xs:string	optional	

Source	`<xs:complexType name="dateDefinition">`
	`  <xs:attribute name="format" type="xs:string"/>`
	`</xs:complexType>`			

Complex Type `datetimeDefinition`

Namespace	No namespace			
Used by	Element	field/datetime-format		
Attributes	**QName**	**Type**	**Use**	
	format	xs:string	optional	

Source	`<xs:complexType name="datetimeDefinition">`
	`  <xs:attribute name="format" type="xs:string"/>`
	`</xs:complexType>`			

Complex Type `attrListDefinition`

Namespace	No namespace			
Used by	Element	field/attr-list-format		
Attributes	**QName**	**Type**	**Use**	
	attr-delimeter	xs:string	required	
	attr-quote	xs:string	required	
	name-value-delimiter	xs:string	optional	
	value-quote	xs:string	optional	

Source	`<xs:complexType name="attrListDefinition">`
	`  <xs:attribute name="attr-delimeter" type="xs:string" use="required"/>`
	`  <xs:attribute name="attr-quote" type="xs:string" use="required"/>`
	`  <xs:attribute name="name-value-delimiter" type="xs:string"/>`
	`  <xs:attribute name="value-quote" type="xs:string"/>`
	`</xs:complexType>`			

Simple Type(s)

Simple Type `emptyRowAction`

Namespace	No namespace	
Type	restriction of xs:string	
Facets	enumeration	stop
	enumeration	ignore
Used by	Attribute	dataSource/@empty-row

Source	<pre><xs:simpleType name="emptyRowAction">
	  <xs:restriction base="xs:string">
	    <xs:enumeration value="stop"/>
	    <xs:enumeration value="ignore"/>
	  </xs:restriction>
	</xs:simpleType></pre>	

Simple Type `fieldType`

Namespace	No namespace	
Type	restriction of xs:string	
Facets	enumeration	bool
	enumeration	int
	enumeration	double
	enumeration	string
	enumeration	text
	enumeration	date
	enumeration	datetime
	enumeration	bool-list
	enumeration	int-list
	enumeration	double-list
	enumeration	string-list
	enumeration	date-list
	enumeration	datetime-list
	enumeration	attribute-list
Used by	Attribute	field/@type

Source	<pre><xs:simpleType name="fieldType">
	  <xs:restriction base="xs:string">
	    <xs:enumeration value="bool"/>
	    <xs:enumeration value="int"/>
	    <xs:enumeration value="double"/>
	    <xs:enumeration value="string"/>
	    <xs:enumeration value="text"/>
	    <xs:enumeration value="date"/>
	    <xs:enumeration value="datetime"/>
	    <xs:enumeration value="bool-list"/>
	    <xs:enumeration value="int-list"/>
	    <xs:enumeration value="double-list"/>
	    <xs:enumeration value="string-list"/>
	    <xs:enumeration value="date-list"/>
	    <xs:enumeration value="datetime-list"/>
	    <xs:enumeration value="attribute-list"/>
	  </xs:restriction>
	</xs:simpleType></pre>	

Simple Type `datasetTarget`

Namespace	No namespace	
Type	restriction of xs:string	
Facets	enumeration	onec-product
	enumeration	onec-group

	enumeration	mage-product
	enumeration	mage-category
	enumeration	mage-price
	enumeration	mage-inventory
	enumeration	mage-user-group
Used by	Attribute	data-set/@target
Source	<xs:simpleType name="datasetTarget"> <xs:restriction base="xs:string"> <xs:enumeration value="onec-product"/> <xs:enumeration value="onec-group"/> <xs:enumeration value="mage-product"/> <xs:enumeration value="mage-category"/> <xs:enumeration value="mage-price"/> <xs:enumeration value="mage-inventory"/> <xs:enumeration value="mage-user-group"/> </xs:restriction> </xs:simpleType>	

Attribute(s)

Attribute `databaseServer / @host`

Namespace	No namespace
Type	xs:string
Properties	use: required
Used by	Complex Type databaseServer
Source	<xs:attribute name="host" type="xs:string" use="required"/>

Attribute `databaseServer / @port`

Namespace	No namespace
Type	xs:int
Properties	content: simple
Used by	Complex Type databaseServer
Source	<xs:attribute name="port" type="xs:int"/>

Attribute `databaseServer / @user`

Namespace	No namespace
Type	xs:string
Properties	use: required
Used by	Complex Type databaseServer
Source	<xs:attribute name="user" type="xs:string" use="required"/>

Attribute `databaseServer / @password`

Namespace	No namespace
Type	xs:string
Properties	use: required
Used by	Complex Type databaseServer
Source	<xs:attribute name="password" type="xs:string" use="required"/>

Attribute `database / @name`

Namespace	No namespace
Annotations	MySql database name

Type	xs:string
Properties	use: required
Used by	Complex Type database

Source	`<xs:attribute name="name" type="xs:string" use="required">`
	  `<xs:annotation>`
	    `<xs:documentation>MySql database name</xs:documentation>`
	  `</xs:annotation>`
	`</xs:attribute>`

Attribute `dataFile / @id`

Namespace	No namespace
Annotations	Data source file id, to use in next configuration
Type	xs:token
Properties	use: required
Used by	Complex Type dataFile

Source	`<xs:attribute name="id" type="xs:token" use="required">`
	  `<xs:annotation>`
	    `<xs:documentation>Data source file id, to use in next configuration</xs:documentation>`
	  `</xs:annotation>`
	`</xs:attribute>`

Attribute `dataFile / @path`

Namespace	No namespace
Annotations	Data source file path
Type	xs:string
Properties	use: required
Used by	Complex Type dataFile

Source	`<xs:attribute name="path" type="xs:string" use="required">`
	  `<xs:annotation>`
	    `<xs:documentation>Data source file path</xs:documentation>`
	  `</xs:annotation>`
	`</xs:attribute>`

Attribute `dataSource / @file`

Namespace	No namespace
Annotations	Reference to file in files section
Type	xs:string
Properties	use: required
Used by	Complex Type dataSource

Source	`<xs:attribute name="file" type="xs:string" use="required">`
	  `<xs:annotation>`
	    `<xs:documentation>Reference to file in files section</xs:documentation>`
	  `</xs:annotation>`
	`</xs:attribute>`

Attribute `dataSource / @sheet`

Namespace	No namespace
Annotations	Name of sheet contains data. For CSV files only one sheet supported - default
Type	xs:string
Properties	use: required
Used by	Complex Type dataSource

Source	`<xs:attribute name="sheet" type="xs:string" use="required">`
	  `<xs:annotation>`
	    `<xs:documentation>Name of sheet contains data. For CSV files only one sheet supported -`
	 `default</xs:documentation>`

```
        </xs:annotation>
    </xs:attribute>
```

Attribute `dataSource / @empty-row`

Namespace	No namespace	
Type	emptyRowAction	
Properties	default:	ignore
Facets	enumeration	stop
	enumeration	ignore
Used by	Complex Type	dataSource
Source	`<xs:attribute name="empty-row" default="ignore" type="emptyRowAction"/>`	

Attribute `listDefinition / @delimeter`

Namespace	No namespace	
Type	xs:string	
Properties	use:	required
Used by	Complex Type	listDefinition
Source	`<xs:attribute name="delimeter" type="xs:string" use="required"/>`	

Attribute `listDefinition / @quote`

Namespace	No namespace	
Type	xs:string	
Properties	use:	required
Used by	Complex Type	listDefinition
Source	`<xs:attribute name="quote" type="xs:string" use="required"/>`	

Attribute `dateDefinition / @format`

Namespace	No namespace	
Type	xs:string	
Properties	content:	simple
Used by	Complex Type	dateDefinition
Source	`<xs:attribute name="format" type="xs:string"/>`	

Attribute `datetimeDefinition / @format`

Namespace	No namespace	
Type	xs:string	
Properties	content:	simple
Used by	Complex Type	datetimeDefinition
Source	`<xs:attribute name="format" type="xs:string"/>`	

Attribute `attrListDefinition / @attr-delimeter`

Namespace	No namespace	
Type	xs:string	
Properties	use:	required
Used by	Complex Type	attrListDefinition
Source	`<xs:attribute name="attr-delimeter" type="xs:string" use="required"/>`	

Attribute `attrListDefinition` / `@attr-quote`

Namespace	No namespace	
Type	xs:string	
Properties	use:	required
Used by	Complex Type	attrListDefinition
Source	`<xs:attribute name="attr-quote" type="xs:string" use="required"/>`	

Attribute `attrListDefinition` / `@name-value-delimiter`

Namespace	No namespace	
Type	xs:string	
Properties	content:	simple
Used by	Complex Type	attrListDefinition
Source	`<xs:attribute name="name-value-delimiter" type="xs:string"/>`	

Attribute `attrListDefinition` / `@value-quote`

Namespace	No namespace	
Type	xs:string	
Properties	content:	simple
Used by	Complex Type	attrListDefinition
Source	`<xs:attribute name="value-quote" type="xs:string"/>`	

Attribute `field` / `@name`

Namespace	No namespace	
Type	xs:string	
Properties	use:	required
Used by	Complex Type	field
Source	`<xs:attribute name="name" type="xs:string" use="required"/>`	

Attribute `field` / `@column`

Namespace	No namespace	
Type	xs:nonNegativeInteger	
Properties	content:	simple
Used by	Complex Type	field
Source	`<xs:attribute name="column" type="xs:nonNegativeInteger"/>`	

Attribute `field` / `@regex`

Namespace	No namespace	
Type	xs:string	
Properties	content:	simple
Used by	Complex Type	field
Source	`<xs:attribute name="regex" type="xs:string"/>`	

Attribute `field` / `@type`

Namespace	No namespace	
Type	fieldType	
Properties	default:	string

Facets	enumeration	bool
	enumeration	int
	enumeration	double
	enumeration	string
	enumeration	text
	enumeration	date
	enumeration	datetime
	enumeration	bool-list
	enumeration	int-list
	enumeration	double-list
	enumeration	string-list
	enumeration	date-list
	enumeration	datetime-list
	enumeration	attribute-list
Used by	Complex Type	field
Source	`<xs:attribute name="type" type="fieldType" default="string"/>`	

Attribute `field` / `@key-field`

Namespace	No namespace	
Type	xs:boolean	
Properties	content:	simple
Used by	Complex Type	field
Source	`<xs:attribute name="key-field" type="xs:boolean"/>`	

Attribute `field` / `@nested`

Namespace	No namespace	
Type	xs:boolean	
Properties	use:	optional
Used by	Complex Type	field
Source	`<xs:attribute name="nested" type="xs:boolean" use="optional"/>`	

Attribute `field` / `@parent`

Namespace	No namespace	
Type	xs:string	
Properties	use:	optional
Used by	Complex Type	field
Source	`<xs:attribute name="parent" type="xs:string" use="optional"/>`	

Attribute `field` / `@optional`

Namespace	No namespace	
Type	xs:boolean	
Properties	default:	false
Used by	Complex Type	field
Source	`<xs:attribute default="false" name="optional" type="xs:boolean"/>`	

Attribute `field` / `@locale`

Namespace	No namespace

Annotations	Language and country code like ru_RU, en_US
Type	xs:string
Properties	content: simple
Used by	Complex Type field

Source	`<xs:attribute name="locale" type="xs:string">`
	`  <xs:annotation>`
	`    <xs:documentation>Language and country code like ru_RU, en_US</xs:documentation>`
	`  </xs:annotation>`
	`</xs:attribute>`

Attribute `fieldSet` / `@name`

Namespace	No namespace
Annotations	Attribute name that should fit to name in data source. This name will be as attribute name in mediation.
Type	xs:string
Properties	use: required
Used by	Complex Type fieldSet

Source	`<xs:attribute name="name" type="xs:string" use="required">`
	`  <xs:annotation>`
	`    <xs:documentation>Attribute name that should fit to name in data source. This name will be as attribute name in mediation.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:attribute>`

Attribute `data-set` / `@name`

Namespace	No namespace
Annotations	Internal dataset id, should be uniqe
Type	xs:string
Properties	use: required
Used by	Complex Type data-set

Source	`<xs:attribute name="name" type="xs:string" use="required">`
	`  <xs:annotation>`
	`    <xs:documentation>Internal dataset id, should be uniqe</xs:documentation>`
	`  </xs:annotation>`
	`</xs:attribute>`

Attribute `data-set` / `@headers-row`

Namespace	No namespace
Annotations	Zero base row number that contains headers. If it is not defined - the data source does not contian headers.
Type	xs:nonNegativeInteger
Properties	content: simple
Used by	Complex Type data-set

Source	`<xs:attribute name="headers-row" type="xs:nonNegativeInteger">`
	`  <xs:annotation>`
	`    <xs:documentation>Zero base row number that contains headers. If it is not defined - the data source does not contian headers.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:attribute>`

Attribute `data-set` / `@rows-to-skip`

Namespace	No namespace
Annotations	Number of rows to skip before real data
Type	xs:int
Properties	content: simple

Used by	Complex Type	data-set

Source	<td colspan="2">`<xs:attribute name="rows-to-skip" type="xs:int">`
	  `<xs:annotation>`
	    `<xs:documentation>Number of rows to skip before real data</xs:documentation>`
	  `</xs:annotation>`
	`</xs:attribute>`</td>

Attribute `data-set / @max-absent-age`

Namespace	No namespace	
Annotations	`Specify number of days to store objects in mediation database that does not exist in data soutrce.`	
Type	xs:int	
Properties	content:	simple
Used by	Complex Type	data-set

Source	<td colspan="2">`<xs:attribute name="max-absent-age" type="xs:int">`
	  `<xs:annotation>`
	    `<xs:documentation>Specify number of days to store objects in mediation database that does not`
	 `exist in data soutrce.</xs:documentation>`
	  `</xs:annotation>`
	`</xs:attribute>`</td>

Attribute `data-set / @target`

Namespace	No namespace	
Annotations	`Destination mediation table`	
Type	datasetTarget	
Properties	use:	required
Facets	enumeration	`onec-product`
	enumeration	`onec-group`
	enumeration	`mage-product`
	enumeration	`mage-category`
	enumeration	`mage-price`
	enumeration	`mage-inventory`
	enumeration	`mage-user-group`
Used by	Complex Type	data-set

Source	<td colspan="2">`<xs:attribute name="target" type="datasetTarget" use="required">`
	  `<xs:annotation>`
	    `<xs:documentation>Destination mediation table</xs:documentation>`
	  `</xs:annotation>`
	`</xs:attribute>`</td>