

# Schema documentation for DR-GW-Management.xsd

november 5, 2024

## Table of Contents

Namespace: "DR-GW-Interface/DR-GW-Management"	2
Schema(s)	2
Main schema DR-GW-Management.xsd	2
Element(s)	2
Element Management_Get	2
Element Management_Post	3
Element Management_Put	3
Element Management_Delete	4
Namespace: "DR-GW-Interface/DR-GW-Management.CommonTypes"	5
Schema(s)	5
Imported schema DR-GW-Management.CommonTypes.xsd	5
Element(s)	5
Element typeManagementRequest / requestUri	5
Element typeManagementRequest / body	6
Element typeManagementRequest / authentication	6
Element typeManagementAuthentication / username	7
Element typeManagementAuthentication / password	7
Element typeManagementRequest / host	7
Element typeManagementEvent / body	7
Complex Type(s)	8
Complex Type typeManagementRequest	8
Complex Type typeManagementAuthentication	9
Complex Type typeManagementEvent	9
Namespace: "DR-GW-Interface/CommonTypes"	10
Schema(s)	10
Imported schema CommonTypes.xsd	10
Element(s)	10
Element ct:typeRequest / ct:requestId	10
Element ct:typeEvent / ct:requestId	10
Element ct:typeEvent / ct:result	10
Element ct:typeResult / ct:responseCode	11
Element ct:typeResult / ct:sourceSystem	11
Element ct:typeResult / ct:result	11
Element ct:typeTSI / ct:mnc	12
Element ct:typeTSI / ct:mcc	12
Element ct:typeTSI / ct:ssi	12
Element ct:typeExternal / ct:gatewayNumber	12
Element ct:typeExternal / ct:number	12
Element ct:typeSubscriberAddress / ct:ssi	13
Element ct:typeSubscriberAddress / ct:tsi	13
Element ct:typeAddress / ct:subscriber	13
Element ct:typeAddress / ct:alias	14
Element ct:typeAddress / ct:msisdn	14
Element ct:typeAddress / ct:fssn	14
Element ct:typeAddress / ct:external	14
Element ct:typeAddress / ct:opta	15
Element ct:typeAddress / ct:cell	15
Element ct:typeResponse / ct:requestId	15
Element ct:typeResponse / ct:result	16
Complex Type(s)	16
Complex Type ct:typeRequest	16
Complex Type ct:typeEvent	16
Complex Type ct:typeResult	17
Complex Type ct:typeTSI	17
Complex Type ct:typeExternal	17
Complex Type ct:typeSubscriberAddress	18
Complex Type ct:typeAddress	18
Complex Type ct:typeResponse	19
Complex Type ct:typeEmpty	19
Simple Type(s)	19

Simple Type ct:typeResponseCode .....	19
Simple Type ct:typeSourceSystem .....	20
Simple Type ct:typeDialString .....	20
Simple Type ct:typeOPTA .....	21
Simple Type ct:typeAddressingStyle .....	21

## Namespace: "DR-GW-Interface/DR-GW-Management"

### Schema(s)

#### Main schema DR-GW-Management.xsd

Namespace	DR-GW-Interface/DR-GW-Management
Annotations	Version 1.2  DR-GW-Management. Use for so called NEM-API(Tactilon-API) functions. Use only via SOAP.
Properties	attribute form default: unqualified element form default: qualified

### Element(s)

#### Element Management\_Get

Namespace	DR-GW-Interface/DR-GW-Management
Diagram	
Type	typeManagementRequest
Type hierarchy	<ul style="list-style-type: none"> <li>ct:typeRequest <ul style="list-style-type: none"> <li>typeManagementRequest</li> </ul> </li> </ul>
Properties	content: complex
Model	ct:requestId, requestUri, body{0,1}, authentication{0,1}, host{0,1}
Children	authentication, body, ct:requestId, host, requestUri

Instance	<pre>&lt;Management_Get xmlns="DR-GW-Interface/DR-GW-Management" xmlns:ct="DR-GW-Interface/ CommonTypes" xmlns:ctM="DR-GW-Interface/DR-GW-Management.CommonTypes"&gt;   &lt;ct:requestId&gt;{1,1}&lt;/ct:requestId&gt;   &lt;ctM:requestUri&gt;{1,1}&lt;/ctM:requestUri&gt;   &lt;ctM:body&gt;{0,1}&lt;/ctM:body&gt;   &lt;ctM:authentication&gt;{0,1}&lt;/ctM:authentication&gt;   &lt;ctM:host&gt;{0,1}&lt;/ctM:host&gt; &lt;/Management_Get&gt;</pre>
Source	<pre>&lt;xs:element name="Management_Get" type="ctM:typeManagementRequest" /&gt;</pre>

## Element Management\_Post

Namespace	DR-GW-Interface/DR-GW-Management
Diagram	
Type	typeManagementRequest
Type hierarchy	<ul style="list-style-type: none"> <li>ct.typeRequest</li> <li>typeManagementRequest</li> </ul>
Properties	content: complex
Model	ct:requestId , requestUri , body{0,1} , authentication{0,1} , host{0,1}
Children	authentication, body, ct:requestId, host, requestUri
Instance	<pre>&lt;Management_Post xmlns="DR-GW-Interface/DR-GW-Management" xmlns:ct="DR-GW-Interface/ CommonTypes" xmlns:ctM="DR-GW-Interface/DR-GW-Management.CommonTypes"&gt;   &lt;ct:requestId&gt;{1,1}&lt;/ct:requestId&gt;   &lt;ctM:requestUri&gt;{1,1}&lt;/ctM:requestUri&gt;   &lt;ctM:body&gt;{0,1}&lt;/ctM:body&gt;   &lt;ctM:authentication&gt;{0,1}&lt;/ctM:authentication&gt;   &lt;ctM:host&gt;{0,1}&lt;/ctM:host&gt; &lt;/Management_Post&gt;</pre>
Source	<pre>&lt;xs:element name="Management_Post" type="ctM:typeManagementRequest" /&gt;</pre>

## Element Management\_Put

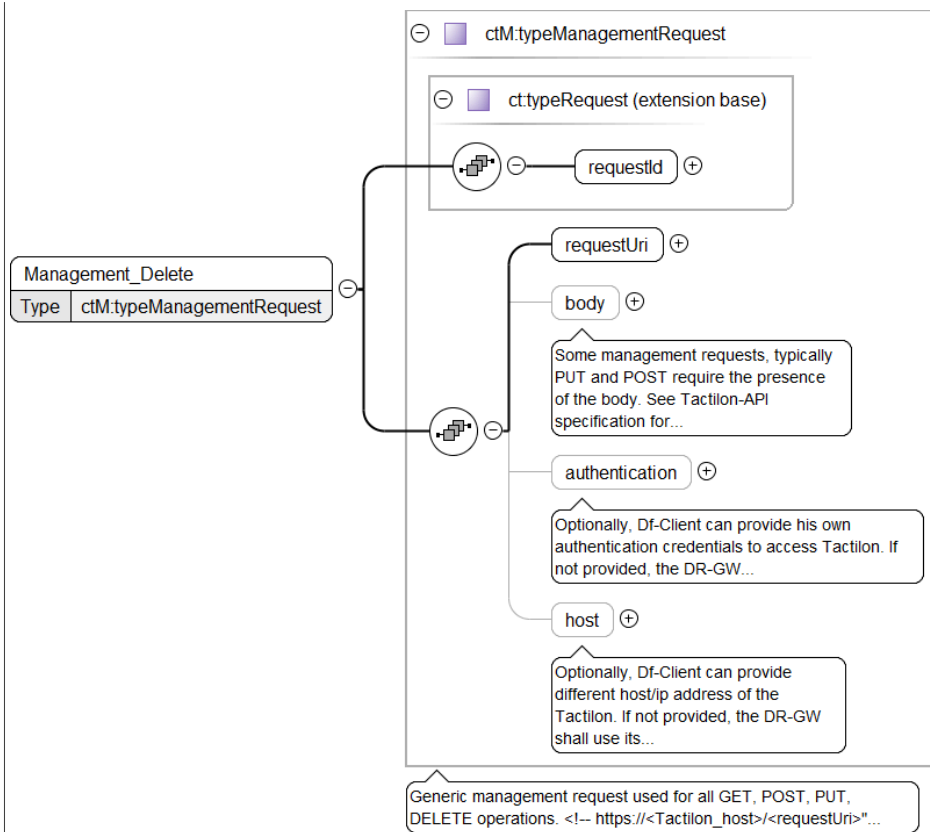
Namespace	DR-GW-Interface/DR-GW-Management
-----------	----------------------------------

Diagram	<p>Management_Put Type ctM:typeManagementRequest</p> <p>ctM:typeManagementRequest</p> <p>ct:typeRequest (extension base)</p> <p>requestId</p> <p>requestUri</p> <p>body</p> <p>Some management requests, typically PUT and POST require the presence of the body. See Tactilon-API specification for...</p> <p>authentication</p> <p>Optionally, Df-Client can provide his own authentication credentials to access Tactilon. If not provided, the DR-GW...</p> <p>host</p> <p>Optionally, Df-Client can provide different host/ip address of the Tactilon. If not provided, the DR-GW shall use its...</p> <p>Generic management request used for all GET, POST, PUT, DELETE operations. &lt;!-- https://&lt;Tactilon_host&gt;/&lt;requestUri&gt;"...</p>
Type	typeManagementRequest
Type hierarchy	<ul style="list-style-type: none"><li>ct:typeRequest</li><li>typeManagementRequest</li></ul>
Properties	content: complex
Model	ct:requestId , requestUri , body{0,1} , authentication{0,1} , host{0,1}
Children	authentication, body, ct:requestId, host, requestUri
Instance	<pre>&lt;Management_Put xmlns="DR-GW-Interface/DR-GW-Management" xmlns:ct="DR-GW-Interface/CommonTypes" xmlns:ctM="DR-GW-Interface/DR-GW-Management.CommonTypes"&gt;   &lt;ct:requestId&gt;{1,1}&lt;/ct:requestId&gt;   &lt;ctM:requestUri&gt;{1,1}&lt;/ctM:requestUri&gt;   &lt;ctM:body&gt;{0,1}&lt;/ctM:body&gt;   &lt;ctM:authentication&gt;{0,1}&lt;/ctM:authentication&gt;   &lt;ctM:host&gt;{0,1}&lt;/ctM:host&gt; &lt;/Management_Put&gt;</pre>
Source	<pre>&lt;xs:element name="Management_Put" type="ctM:typeManagementRequest" /&gt;</pre>

Element Management\_Delete

Namespace	DR-GW-Interface/DR-GW-Management
-----------	----------------------------------

Diagram



Type	typeManagementRequest
Type hierarchy	<ul style="list-style-type: none"> <li>ct:typeRequest</li> <li>typeManagementRequest</li> </ul>
Properties	content: complex
Model	ct:requestId, requestUri, body{0,1}, authentication{0,1}, host{0,1}
Children	authentication, body, ct:requestId, host, requestUri
Instance	<pre>&lt;Management_Delete xmlns="DR-GW-Interface/DR-GW-Management" xmlns:ct="DR-GW-Interface/CommonTypes" xmlns:ctM="DR-GW-Interface/DR-GW-Management.CommonTypes"&gt;   &lt;ct:requestId&gt;{1,1}&lt;/ct:requestId&gt;   &lt;ctM:requestUri&gt;{1,1}&lt;/ctM:requestUri&gt;   &lt;ctM:body&gt;{0,1}&lt;/ctM:body&gt;   &lt;ctM:authentication&gt;{0,1}&lt;/ctM:authentication&gt;   &lt;ctM:host&gt;{0,1}&lt;/ctM:host&gt; &lt;/Management_Delete&gt;</pre>
Source	<xs:element name="Management_Delete" type="ctM:typeManagementRequest"/>

**Namespace: "DR-GW-Interface/DR-GW-Management.CommonTypes"**

**Schema(s)**

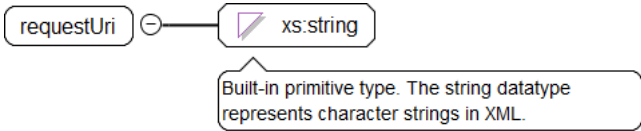
**Imported schema DR-GW-Management.CommonTypes.xsd**

Namespace	DR-GW-Interface/DR-GW-Management.CommonTypes
Properties	attribute form default: unqualified element form default: qualified

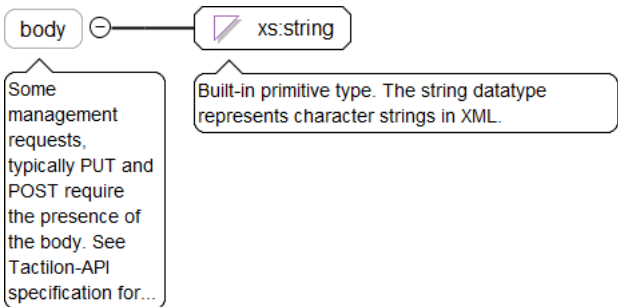
**Element(s)**

**Element typeManagementRequest / requestUri**

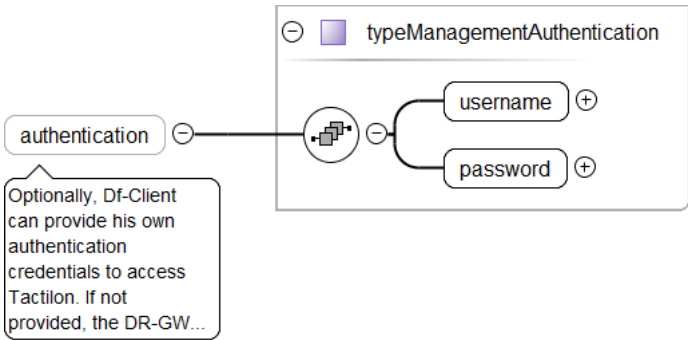
Namespace	DR-GW-Interface/DR-GW-Management.CommonTypes
-----------	--

Diagram	
Type	xs:string
Properties	content: simple
Source	<code>&lt;xs:element name="requestUri" type="xs:string"/&gt;</code>

## Element typeManagementRequest / body

Namespace	DR-GW-Interface/DR-GW-Management.CommonTypes
Annotations	Some management requests, typically PUT and POST require the presence of the body. See Tactilon-API specification for details on how to construct the body.
Diagram	
Type	xs:string
Properties	content: simple minOccurs: 0
Source	<pre> &lt;xs:element name="body" type="xs:string" minOccurs="0"&gt;   &lt;xs:annotation&gt;     &lt;xs:documentation&gt;Some management requests, typically PUT and POST require the presence of the       body. See Tactilon-API specification for details on how to construct the body.&lt;/xs:documentation&gt;   &lt;/xs:annotation&gt; &lt;/xs:element&gt; </pre>

## Element typeManagementRequest / authentication

Namespace	DR-GW-Interface/DR-GW-Management.CommonTypes
Annotations	Optionally, Df-Client can provide his own authentication credentials to access Tactilon. If not provided, the DR-GW shall use its own, globally configured.
Diagram	
Type	typeManagementAuthentication
Properties	content: complex minOccurs: 0
Model	username , password
Children	password, username
Instance	<code>&lt;authentication xmlns="DR-GW-Interface/DR-GW-Management.CommonTypes"&gt;</code>

	<pre>&lt;username&gt;{1,1}&lt;/username&gt; &lt;password&gt;{1,1}&lt;/password&gt; &lt;/authentication&gt;</pre>
Source	<pre>&lt;xs:element name="authentication" type="typeManagementAuthentication" minOccurs="0"&gt;   &lt;xs:annotation&gt;     &lt;xs:documentation&gt;Optionally, Df-Client can provide his own authentication credentials to access     Tactilon. If not provided, the DR-GW shall use its own, globally configured.&lt;/xs:documentation&gt;   &lt;/xs:annotation&gt; &lt;/xs:element&gt;</pre>

### Element typeManagementAuthentication / username

Namespace	DR-GW-Interface/DR-GW-Management.CommonTypes
Diagram	
Type	xs:string
Properties	content: simple
Source	<pre>&lt;xs:element name="username" type="xs:string" /&gt;</pre>

### Element typeManagementAuthentication / password

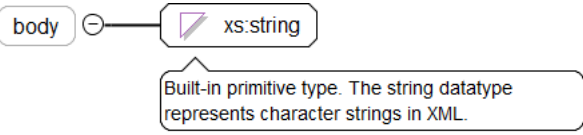
Namespace	DR-GW-Interface/DR-GW-Management.CommonTypes
Diagram	
Type	xs:string
Properties	content: simple
Source	<pre>&lt;xs:element name="password" type="xs:string" /&gt;</pre>

### Element typeManagementRequest / host

Namespace	DR-GW-Interface/DR-GW-Management.CommonTypes
Annotations	Optionally, Df-Client can provide different host/ip address of the Tactilon. If not provided, the DR-GW shall use its own, globally configured.
Diagram	
Type	xs:string
Properties	content: simple minOccurs: 0
Source	<pre>&lt;xs:element name="host" type="xs:string" minOccurs="0"&gt;   &lt;xs:annotation&gt;     &lt;xs:documentation&gt;Optionally, Df-Client can provide different host/ip address of the Tactilon.     If not provided, the DR-GW shall use its own, globally configured.&lt;/xs:documentation&gt;   &lt;/xs:annotation&gt; &lt;/xs:element&gt;</pre>

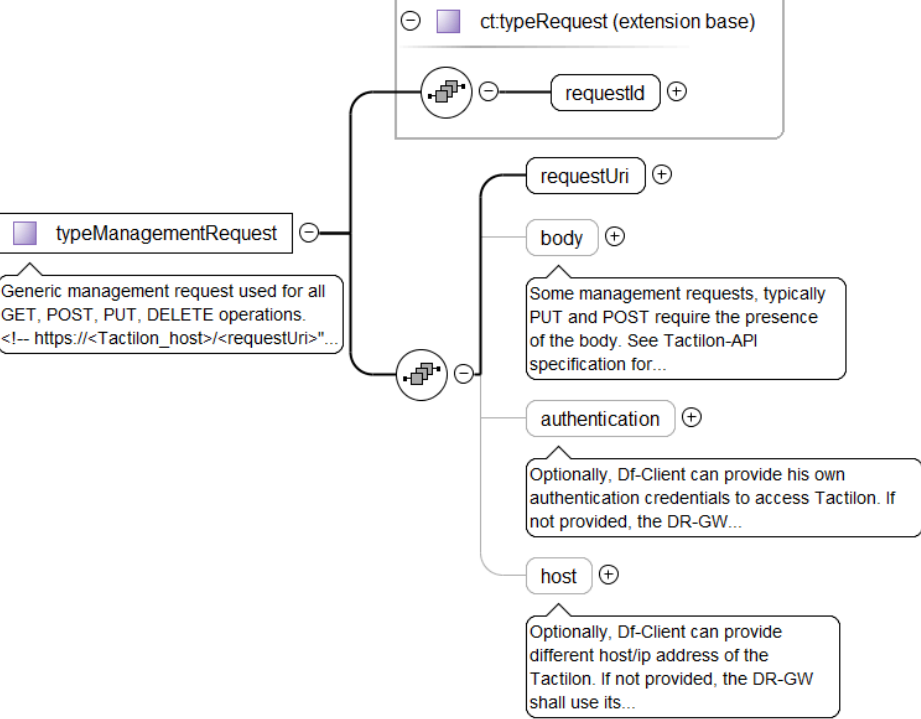
### Element typeManagementEvent / body

Namespace	DR-GW-Interface/DR-GW-Management.CommonTypes
-----------	--

Diagram	 <p>body — xs:string</p> <p>Built-in primitive type. The string datatype represents character strings in XML.</p>				
Type	xs:string				
Properties	<table> <tr> <td>content:</td><td>simple</td></tr> <tr> <td>minOccurs:</td><td>0</td></tr> </table>	content:	simple	minOccurs:	0
content:	simple				
minOccurs:	0				
Source	<code>&lt;xs:element name="body" type="xs:string" minOccurs="0"/&gt;</code>				

## Complex Type(s)

### Complex Type typeManagementRequest

Namespace	DR-GW-Interface/DR-GW-Management.CommonTypes
Annotations	Generic management request used for all GET, POST, PUT, DELETE operations.
Diagram	 <p>typeManagementRequest</p> <p>Generic management request used for all GET, POST, PUT, DELETE operations. &lt;!-- https://&lt;Tactilon_host&gt;/&lt;requestUri&gt;"...</p> <p>ct.typeRequest (extension base)</p> <p>requestId</p> <p>requestUri</p> <p>body</p> <p>Some management requests, typically PUT and POST require the presence of the body. See Tactilon-API specification for...</p> <p>authentication</p> <p>Optionally, Df-Client can provide his own authentication credentials to access Tactilon. If not provided, the DR-GW...</p> <p>host</p> <p>Optionally, Df-Client can provide different host/ip address of the Tactilon. If not provided, the DR-GW shall use its...</p>
Type	extension of ct:typeRequest
Type hierarchy	<ul style="list-style-type: none"> <li>ct:typeRequest</li> <li>typeManagementRequest</li> </ul>
Used by	Elements Management_Delete, Management_Get, Management_Post, Management_Put
Model	ct:requestId, requestUri, body{0,1}, authentication{0,1}, host{0,1}
Children	authentication, body, ct:requestId, host, requestUri
Source	<pre> &lt;xs:complexType name="typeManagementRequest"&gt;   &lt;xs:annotation&gt;     &lt;xs:documentation&gt;Generic management request used for all GET, POST, PUT, DELETE operations.     &lt;!-- https://&lt;Tactilon_host&gt;/&lt;requestUri&gt;" --&gt;   &lt;/xs:documentation&gt; &lt;/xs:annotation&gt;   &lt;xs:complexContent&gt;     &lt;xs:extension base="ct:typeRequest"&gt;       &lt;xs:sequence&gt;         &lt;xs:element name="requestUri" type="xs:string"/&gt;         &lt;xs:element name="body" type="xs:string" minOccurs="0"&gt;           &lt;xs:documentation&gt;Some management requests, typically PUT and POST require the presence of the body. See Tactilon-API specification for details on how to construct the body.&lt;/ xs:documentation&gt;         &lt;/xs:element&gt;         &lt;xs:element name="authentication" type="xs:string" minOccurs="0"&gt;           &lt;xs:documentation&gt;Optionally, Df-Client can provide his own authentication credentials to access Tactilon. If not provided, the DR-GW shall use its...&lt;/xs:documentation&gt;         &lt;/xs:element&gt;         &lt;xs:element name="host" type="xs:string" minOccurs="0"&gt;           &lt;xs:documentation&gt;Optionally, Df-Client can provide different host/ip address of the Tactilon. If not provided, the DR-GW shall use its...&lt;/xs:documentation&gt;         &lt;/xs:element&gt;       &lt;/xs:sequence&gt;     &lt;/xs:extension&gt;   &lt;/xs:complexContent&gt; &lt;/xs:complexType&gt; </pre>



```

</xs:annotation>
</xs:element>
<xs:element name="authentication" type="typeManagementAuthentication" minOccurs="0">
  <xs:annotation>
    <xs:documentation>Optionally, Df-Client can provide his own authentication credentials
to access Tactilon. If not provided, the DR-GW shall use its own, globally configured.</
xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="host" type="xs:string" minOccurs="0">
  <xs:annotation>
    <xs:documentation>Optionally, Df-Client can provide different host/ip address of the
Tactilon. If not provided, the DR-GW shall use its own, globally configured.</xs:documentation>
  </xs:annotation>
</xs:element>
</xs:sequence>
</xs:extension>
</xs:complexContent>
</xs:complexType>

```

## Complex Type typeManagementAuthentication

Namespace	DR-GW-Interface/DR-GW-Management.CommonTypes
Annotations	
Diagram	
Used by	Element typeManagementRequest/authentication
Model	username , password
Children	password, username
Source	<pre> &lt;xs:complexType name="typeManagementAuthentication"&gt;   &lt;xs:annotation&gt;     &lt;xs:documentation&gt;     &lt;/xs:documentation&gt;   &lt;/xs:annotation&gt;   &lt;xs:sequence&gt;     &lt;xs:element name="username" type="xs:string"/&gt;     &lt;xs:element name="password" type="xs:string"/&gt;   &lt;/xs:sequence&gt; &lt;/xs:complexType&gt; </pre>

## Complex Type typeManagementEvent

Namespace	DR-GW-Interface/DR-GW-Management.CommonTypes
Annotations	
Diagram	
Type	extension of ct:typeEvent
Type hierarchy	<ul style="list-style-type: none"> <li>ct:typeEvent</li> <li>typeManagementEvent</li> </ul>
Model	ct:requestId{0,1} , ct:result{0,1} , body{0,1}
Children	body, ct:requestId, ct:result
Source	<pre> &lt;xs:complexType name="typeManagementEvent"&gt;   &lt;xs:annotation&gt;     &lt;xs:documentation/&gt;   &lt;/xs:annotation&gt;   &lt;xs:complexContent&gt;     &lt;xs:extension base="ct:typeEvent"&gt; </pre>

```
<xs:sequence>
  <xs:element name="body" type="xs:string" minOccurs="0" />
</xs:sequence>
</xs:extension>
</xs:complexContent>
</xs:complexType>
```

## Namespace: "DR-GW-Interface/CommonTypes"

### Schema(s)

#### Imported schema `CommonTypes.xsd`

Namespace	DR-GW-Interface/CommonTypes
Annotations	Version 1.2
Properties	attribute form default: unqualified
	element form default: qualified

### Element(s)

#### Element `ct:typeRequest` / `ct:requestId`

Namespace	DR-GW-Interface/CommonTypes
Diagram	
Type	xs:unsignedLong
Properties	content: simple
Source	<code>&lt;xs:element name="requestId" type="xs:unsignedLong" /&gt;</code>

#### Element `ct:typeEvent` / `ct:requestId`

Namespace	DR-GW-Interface/CommonTypes
Diagram	
Type	xs:unsignedLong
Properties	content: simple
	minOccurs: 0
Source	<code>&lt;xs:element name="requestId" type="xs:unsignedLong" minOccurs="0" /&gt;</code>

#### Element `ct:typeEvent` / `ct:result`

Namespace	DR-GW-Interface/CommonTypes
Diagram	

Type	ct:typeResult
Properties	content: complex
	minOccurs: 0
Model	ct:responseCode , ct:sourceSystem{0,1} , ct:result{0,1}
Children	ct:responseCode, ct:result, ct:sourceSystem
Instance	<pre>&lt;ct:result xmlns:ct="DR-GW-Interface/CommonTypes"&gt;   &lt;ct:responseCode&gt;{1,1}&lt;/ct:responseCode&gt;   &lt;ct:sourceSystem&gt;{0,1}&lt;/ct:sourceSystem&gt;   &lt;ct:result&gt;{0,1}&lt;/ct:result&gt; &lt;/ct:result&gt;</pre>
Source	<pre>&lt;xs:element name="result" type="ct:typeResult" minOccurs="0"/&gt;</pre>

### Element ct:responseCode / ct:responseCode

Namespace	DR-GW-Interface/CommonTypes
Diagram	
Type	ct:typeResponseCode
Properties	content: simple
Facets	enumeration success
	enumeration final_response_pending
	enumeration error
	enumeration not_authorized_error
	enumeration temporary_failure
	enumeration subscription_failed
Source	<pre>&lt;xs:element name="responseCode" type="ct:typeResponseCode"/&gt;</pre>

### Element ct:typeResult / ct:sourceSystem

Namespace	DR-GW-Interface/CommonTypes
Diagram	
Type	ct:typeSourceSystem
Properties	content: simple
	minOccurs: 0
Facets	enumeration DR-GW
	enumeration TCS-API
	enumeration TETRA
	enumeration TACTILON-API
Source	<pre>&lt;xs:element name="sourceSystem" type="ct:typeSourceSystem" minOccurs="0"/&gt;</pre>

### Element ct:typeResult / ct:result

Namespace	DR-GW-Interface/CommonTypes
Diagram	
Type	xs:unsignedLong
Properties	content: simple
	minOccurs: 0
Source	<pre>&lt;xs:element name="result" type="xs:unsignedLong" minOccurs="0"/&gt;</pre>

## Element `ct:typeTSI` / `ct:mnc`

Namespace	DR-GW-Interface/CommonTypes
Diagram	
Type	xs:unsignedShort
Properties	content: simple
Source	<code>&lt;xs:element name="mnc" type="xs:unsignedShort"/&gt;</code>

## Element `ct:typeTSI` / `ct:mcc`

Namespace	DR-GW-Interface/CommonTypes
Diagram	
Type	xs:unsignedShort
Properties	content: simple
Source	<code>&lt;xs:element name="mcc" type="xs:unsignedShort"/&gt;</code>

## Element `ct:typeTSI` / `ct:ssi`

Namespace	DR-GW-Interface/CommonTypes
Diagram	
Type	xs:unsignedLong
Properties	content: simple
Source	<code>&lt;xs:element name="ssi" type="xs:unsignedLong"/&gt;</code>

## Element `ct:typeExternal` / `ct:gatewayNumber`

Namespace	DR-GW-Interface/CommonTypes
Diagram	
Type	xs:unsignedLong
Properties	content: simple
Source	<code>&lt;xs:element name="gatewayNumber" type="xs:unsignedLong"/&gt;</code>

## Element `ct:typeExternal` / `ct:number`

Namespace	DR-GW-Interface/CommonTypes
Diagram	

Type	ct:typeDialString
Properties	content: simple
Facets	maxLength 24
Source	<code>&lt;xs:element name="number" type="ct:typeDialString"/&gt;</code>

### Element ct:typeSubscriberAddress / ct:ssi

Namespace	DR-GW-Interface/CommonTypes
Diagram	
Type	xs:unsignedLong
Properties	content: simple
Source	<code>&lt;xs:element name="ssi" type="xs:unsignedLong"/&gt;</code>

### Element ct:typeSubscriberAddress / ct:tsi

Namespace	DR-GW-Interface/CommonTypes
Diagram	
Type	ct:typeTSI
Properties	content: complex
Model	ct:mnc , ct:mcc , ct:ssi
Children	ct:mcc, ct:mnc, ct:ssi
Instance	<pre>&lt;ct:tsi xmlns:ct="DR-GW-Interface/CommonTypes"&gt;   &lt;ct:mnc&gt;{1,1}&lt;/ct:mnc&gt;   &lt;ct:mcc&gt;{1,1}&lt;/ct:mcc&gt;   &lt;ct:ssi&gt;{1,1}&lt;/ct:ssi&gt; &lt;/ct:tsi&gt;</pre>
Source	<code>&lt;xs:element name="tsi" type="ct:typeTSI"/&gt;</code>

### Element ct:typeAddress / ct:subscriber

Namespace	DR-GW-Interface/CommonTypes
Diagram	
Type	ct:typeSubscriberAddress
Properties	content: complex minOccurs: 0

Model	ct:ssi   ct:tsi
Children	ct:ssi, ct:tsi
Instance	<pre>&lt;ct:subscriber xmlns:ct="DR-GW-Interface/CommonTypes"&gt;   &lt;ct:ssi&gt;{1,1}&lt;/ct:ssi&gt;   &lt;ct:tsi&gt;{1,1}&lt;/ct:tsi&gt; &lt;/ct:subscriber&gt;</pre>
Source	<pre>&lt;xs:element name="subscriber" type="ct:typeSubscriberAddress" minOccurs="0"/&gt;</pre>

## Element ct:typeAddress / ct:alias

Namespace	DR-GW-Interface/CommonTypes				
Diagram					
Type	xs:normalizedString				
Properties	<table> <tr> <td>content:</td><td>simple</td></tr> <tr> <td>minOccurs:</td><td>0</td></tr> </table>	content:	simple	minOccurs:	0
content:	simple				
minOccurs:	0				
Source	<pre>&lt;xs:element name="alias" type="xs:normalizedString" minOccurs="0"/&gt;</pre>				

## Element ct:typeAddress / ct:msisdn

Namespace	DR-GW-Interface/CommonTypes				
Diagram					
Type	ct:typeDialString				
Properties	<table> <tr> <td>content:</td><td>simple</td></tr> <tr> <td>minOccurs:</td><td>0</td></tr> </table>	content:	simple	minOccurs:	0
content:	simple				
minOccurs:	0				
Facets	<table> <tr> <td>maxLength</td><td>24</td></tr> </table>	maxLength	24		
maxLength	24				
Source	<pre>&lt;xs:element name="msisdn" type="ct:typeDialString" minOccurs="0"/&gt;</pre>				

## Element ct:typeAddress / ct:fssn

Namespace	DR-GW-Interface/CommonTypes				
Annotations	Fleet specific short number				
Diagram					
Type	xs:unsignedLong				
Properties	<table> <tr> <td>content:</td><td>simple</td></tr> <tr> <td>minOccurs:</td><td>0</td></tr> </table>	content:	simple	minOccurs:	0
content:	simple				
minOccurs:	0				
Source	<pre>&lt;xs:element name="fssn" type="xs:unsignedLong" minOccurs="0"&gt;   &lt;xs:annotation&gt;     &lt;xs:documentation&gt;Fleet specific short number&lt;/xs:documentation&gt;   &lt;/xs:annotation&gt; &lt;/xs:element&gt;</pre>				

## Element ct:typeAddress / ct:external

Namespace	DR-GW-Interface/CommonTypes
-----------	-----------------------------

Diagram					
Type	ct:typeExternal				
Properties	<table> <tr> <td>content:</td><td>complex</td></tr> <tr> <td>minOccurs:</td><td>0</td></tr> </table>	content:	complex	minOccurs:	0
content:	complex				
minOccurs:	0				
Model	ct:gatewayNumber , ct:number				
Children	ct:gatewayNumber, ct:number				
Instance	<pre>&lt;ct:external xmlns:ct="DR-GW-Interface/CommonTypes"&gt;   &lt;ct:gatewayNumber&gt;{1,1}&lt;/ct:gatewayNumber&gt;   &lt;ct:number&gt;{1,1}&lt;/ct:number&gt; &lt;/ct:external&gt;</pre>				
Source	<pre>&lt;xs:element name="external" type="ct:typeExternal" minOccurs="0" /&gt;</pre>				

### Element ct:typeAddress / ct:opta

Namespace	DR-GW-Interface/CommonTypes				
Diagram					
Type	ct:typeOPTA				
Properties	<table> <tr> <td>content:</td><td>simple</td></tr> <tr> <td>minOccurs:</td><td>0</td></tr> </table>	content:	simple	minOccurs:	0
content:	simple				
minOccurs:	0				
Facets	<table> <tr> <td>maxLength</td><td>24</td></tr> </table>	maxLength	24		
maxLength	24				
Source	<pre>&lt;xs:element name="opta" type="ct:typeOPTA" minOccurs="0" /&gt;</pre>				

### Element ct:typeAddress / ct:cell

Namespace	DR-GW-Interface/CommonTypes				
Diagram					
Type	xs:short				
Properties	<table> <tr> <td>content:</td><td>simple</td></tr> <tr> <td>minOccurs:</td><td>0</td></tr> </table>	content:	simple	minOccurs:	0
content:	simple				
minOccurs:	0				
Source	<pre>&lt;xs:element name="cell" type="xs:short" minOccurs="0" /&gt;</pre>				

### Element ct:typeResponse / ct:requestId

Namespace	DR-GW-Interface/CommonTypes
Diagram	
Type	xs:unsignedLong

Properties	content: simple
Source	<code>&lt;xs:element name="requestId" type="xs:unsignedLong" /&gt;</code>

## Element `ct:typeResponse` / `ct:result`

Namespace	DR-GW-Interface/CommonTypes
Diagram	
Type	<code>ct:typeResult</code>
Properties	content: complex
Model	<code>ct:responseCode</code> , <code>ct:sourceSystem</code> {0,1} , <code>ct:result</code> {0,1}
Children	<code>ct:responseCode</code> , <code>ct:result</code> , <code>ct:sourceSystem</code>
Instance	<pre>&lt;ct:result xmlns:ct="DR-GW-Interface/CommonTypes"&gt;   &lt;ct:responseCode&gt;{1,1}&lt;/ct:responseCode&gt;   &lt;ct:sourceSystem&gt;{0,1}&lt;/ct:sourceSystem&gt;   &lt;ct:result&gt;{0,1}&lt;/ct:result&gt; &lt;/ct:result&gt;</pre>
Source	<code>&lt;xs:element name="result" type="ct:typeResult" /&gt;</code>

## Complex Type(s)

### Complex Type `ct:typeRequest`

Namespace	DR-GW-Interface/CommonTypes
Diagram	
Used by	Complex Type <code>typeManagementRequest</code>
Model	<code>ct:requestId</code>
Children	<code>ct:requestId</code>
Source	<pre>&lt;xs:complexType name="typeRequest"&gt;   &lt;xs:sequence&gt;     &lt;xs:element name="requestId" type="xs:unsignedLong" /&gt;   &lt;/xs:sequence&gt; &lt;/xs:complexType&gt;</pre>

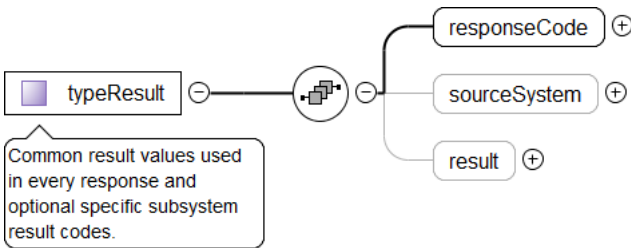
### Complex Type `ct:typeEvent`

Namespace	DR-GW-Interface/CommonTypes
Diagram	
Used by	Complex Type <code>typeManagementEvent</code>
Model	<code>ct:requestId</code> {0,1} , <code>ct:result</code> {0,1}
Children	<code>ct:requestId</code> , <code>ct:result</code>
Source	<pre>&lt;xs:complexType name="typeEvent"&gt;   &lt;xs:sequence&gt;     &lt;xs:element name="requestId" type="xs:unsignedLong" minOccurs="0" /&gt;   &lt;/xs:sequence&gt; &lt;/xs:complexType&gt;</pre>

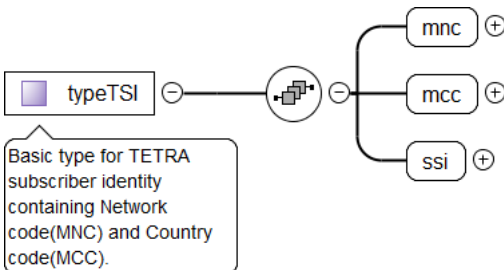


```
<xs:element name="result" type="ct:typeResult" minOccurs="0"/>
</xs:sequence>
</xs:complexType>
```

## Complex Type ct:typeResult

Namespace	DR-GW-Interface/CommonTypes
Annotations	Common result values used in every response and optional specific subsystem result codes.
Diagram	
Used by	Elements ct:typeEvent/ct:result, ct:typeResponse/ct:result
Model	ct:responseCode , ct:sourceSystem{0,1} , ct:result{0,1}
Children	ct:responseCode, ct:result, ct:sourceSystem
Source	<pre>&lt;xs:complexType name="typeResult"&gt;   &lt;xs:annotation&gt;     &lt;xs:documentation&gt;Common result values used in every response and optional specific subsystem     result codes.&lt;/xs:documentation&gt;   &lt;/xs:annotation&gt;   &lt;xs:sequence&gt;     &lt;xs:element name="responseCode" type="ct:typeResponseCode"/&gt;     &lt;xs:element name="sourceSystem" type="ct:typeSourceSystem" minOccurs="0"/&gt;     &lt;xs:element name="result" type="xs:unsignedLong" minOccurs="0"/&gt;   &lt;/xs:sequence&gt; &lt;/xs:complexType&gt;</pre>

## Complex Type ct:typeTSI

Namespace	DR-GW-Interface/CommonTypes
Annotations	Basic type for TETRA subscriber identity containing Network code(MNC) and Country code(MCC).
Diagram	
Used by	Element ct:typeSubscriberAddress/ct:tsi
Model	ct:mnc , ct:mcc , ct:ssi
Children	ct:mcc, ct:mnc, ct:ssi
Source	<pre>&lt;xs:complexType name="typeTSI"&gt;   &lt;xs:annotation&gt;     &lt;xs:documentation&gt;Basic type for TETRA subscriber identity containing Network code(MNC) and     Country code(MCC).&lt;/xs:documentation&gt;   &lt;/xs:annotation&gt;   &lt;xs:sequence&gt;     &lt;xs:element name="mnc" type="xs:unsignedShort"/&gt;     &lt;xs:element name="mcc" type="xs:unsignedShort"/&gt;     &lt;xs:element name="ssi" type="xs:unsignedLong"/&gt;   &lt;/xs:sequence&gt; &lt;/xs:complexType&gt;</pre>

## Complex Type ct:typeExternal

Namespace	DR-GW-Interface/CommonTypes
Annotations	External number consisting of Gateway number + DialString

Diagram	
Used by	Element ct:typeAddress/ct:external
Model	ct:gatewayNumber , ct:number
Children	ct:gatewayNumber, ct:number
Source	<pre> &lt;xs:complexType name="typeExternal"&gt;   &lt;xs:annotation&gt;     &lt;xs:documentation&gt;External number consisting of Gateway number + DialString&lt;/xs:documentation&gt;   &lt;/xs:annotation&gt;   &lt;xs:sequence&gt;     &lt;xs:element name="gatewayNumber" type="xs:unsignedLong" /&gt;     &lt;xs:element name="number" type="ct:typeDialString" /&gt;   &lt;/xs:sequence&gt; &lt;/xs:complexType&gt; </pre>

### Complex Type ct:typeSubscriberAddress

Namespace	DR-GW-Interface/CommonTypes
Annotations	
Diagram	
Used by	Element ct:typeAddress/ct:subscriber
Model	ct:ssi   ct:tsi
Children	ct:ssi, ct:tsi
Source	<pre> &lt;xs:complexType name="typeSubscriberAddress"&gt;   &lt;xs:annotation&gt;     &lt;xs:documentation/&gt;   &lt;/xs:annotation&gt;   &lt;xs:choice&gt;     &lt;xs:element name="ssi" type="xs:unsignedLong" /&gt;     &lt;xs:element name="tsi" type="ct:typeTSI" /&gt;   &lt;/xs:choice&gt; &lt;/xs:complexType&gt; </pre>

### Complex Type ct:typeAddress

Namespace	DR-GW-Interface/CommonTypes
Annotations	Basic type for all possible TETRA address types (SSI, TSI, MSISDN, FSSN, OPTA).
Diagram	
Model	ct:subscriber{0,1} , ct:alias{0,1} , ct:msisdn{0,1} , ct:fssn{0,1} , ct:external{0,1} , ct:opta{0,1} , ct:cell{0,1}

Children	ct:alias, ct:cell, ct:external, ct:fssn, ct:msisdn, ct:opta, ct:subscriber
Source	<pre> &lt;xs:complexType name="typeAddress"&gt;   &lt;xs:annotation&gt;     &lt;xs:documentation&gt;Basic type for all possible TETRA address types (SSI, TSI, MSISDN, FSSN, OPTA).&lt;/xs:documentation&gt;   &lt;/xs:annotation&gt;   &lt;xs:sequence&gt;     &lt;xs:element name="subscriber" type="ct:typeSubscriberAddress" minOccurs="0"/&gt;     &lt;xs:element name="alias" type="xs:normalizedString" minOccurs="0"/&gt;     &lt;xs:element name="msisdn" type="ct:typeDialString" minOccurs="0"/&gt;     &lt;xs:element name="fssn" type="xs:unsignedLong" minOccurs="0"&gt;       &lt;xs:annotation&gt;         &lt;xs:documentation&gt;Fleet specific short number&lt;/xs:documentation&gt;       &lt;/xs:annotation&gt;     &lt;/xs:element&gt;     &lt;xs:element name="external" type="ct:typeExternal" minOccurs="0"/&gt;     &lt;xs:element name="opta" type="ct:typeOPTA" minOccurs="0"/&gt;     &lt;xs:element name="cell" type="xs:short" minOccurs="0"/&gt;   &lt;/xs:sequence&gt; &lt;/xs:complexType&gt; </pre>

## Complex Type ct:typeResponse

Namespace	DR-GW-Interface/CommonTypes
Annotations	Response contains result of execution of any method.
Diagram	
Model	ct:requestId, ct:result
Children	ct:requestId, ct:result
Source	<pre> &lt;xs:complexType name="typeResponse"&gt;   &lt;xs:annotation&gt;     &lt;xs:documentation&gt;Response contains result of execution of any method.&lt;/xs:documentation&gt;   &lt;/xs:annotation&gt;   &lt;xs:sequence&gt;     &lt;xs:element name="requestId" type="xs:unsignedLong"/&gt;     &lt;xs:element name="result" type="ct:typeResult"/&gt;   &lt;/xs:sequence&gt; &lt;/xs:complexType&gt; </pre>

## Complex Type ct:typeEmpty

Namespace	DR-GW-Interface/CommonTypes
Annotations	Explicit type specification for elements that shall be empty.
Diagram	
Source	<pre> &lt;xs:complexType name="typeEmpty"&gt;   &lt;xs:annotation&gt;     &lt;xs:documentation&gt;Explicit type specification for elements that shall be empty.&lt;/xs:documentation&gt;   &lt;/xs:annotation&gt; &lt;/xs:complexType&gt; </pre>

## Simple Type(s)

### Simple Type ct:typeResponseCode

Namespace	DR-GW-Interface/CommonTypes
Diagram	

Type	restriction of xs:normalizedString	
Facets	enumeration	success
	enumeration	final_response_pending
	enumeration	error
	enumeration	not_authorized_error
	enumeration	temporary_failure
	enumeration	subscription_failed
Used by	Element	ct:typeResult/ct:responseCode
Source	<pre> &lt;xs:simpleType name="typeResponseCode"&gt;   &lt;xs:restriction base="xs:normalizedString"&gt;     &lt;xs:enumeration value="success"/&gt;     &lt;xs:enumeration value="final_response_pending"/&gt;     &lt;xs:enumeration value="error"/&gt;     &lt;xs:enumeration value="not_authorized_error"/&gt;     &lt;xs:enumeration value="temporary_failure"/&gt;     &lt;xs:enumeration value="subscription_failed"/&gt;   &lt;/xs:restriction&gt; &lt;/xs:simpleType&gt; </pre>	

### Simple Type ct:typeSourceSystem

Namespace	DR-GW-Interface/CommonTypes	
Diagram		
Type	restriction of xs:normalizedString	
Facets	enumeration	DR-GW
	enumeration	TCS-API
	enumeration	TETRA
	enumeration	TACTILON-API
Used by	Element	ct:typeResult/ct:sourceSystem
Source	<pre> &lt;xs:simpleType name="typeSourceSystem"&gt;   &lt;xs:restriction base="xs:normalizedString"&gt;     &lt;xs:enumeration value="DR-GW"/&gt;     &lt;xs:enumeration value="TCS-API"/&gt;     &lt;xs:enumeration value="TETRA"/&gt;     &lt;xs:enumeration value="TACTILON-API"/&gt;   &lt;/xs:restriction&gt; &lt;/xs:simpleType&gt; </pre>	

### Simple Type ct:typeDialString

Namespace	DR-GW-Interface/CommonTypes	
Annotations	Allowed characters are digits 0 - 9, *, #, A, B, C and D. Maximum length is 24 characters.	
Diagram		
Type	restriction of xs:normalizedString	
Facets	maxLength	24
Used by	Elements	ct:typeAddress/ct:msisdn, ct:typeExternal/ct:number
Source	<pre> &lt;xs:simpleType name="typeDialString"&gt;   &lt;xs:annotation&gt;     &lt;xs:documentation&gt;Allowed characters are digits 0 - 9, *, #, A, B, C and D. Maximum length is 24 characters.&lt;/xs:documentation&gt;   &lt;/xs:annotation&gt;   &lt;xs:restriction base="xs:normalizedString"&gt;     &lt;xs:maxLength value="24"/&gt;   &lt;/xs:restriction&gt; &lt;/xs:simpleType&gt; </pre>	

```
</xs:annotation>
<xs:restriction base="xs:normalizedString">
  <xs:maxLength value="24"/>
</xs:restriction>
</xs:simpleType>
```

## Simple Type ct:typeOPTA

Namespace	DR-GW-Interface/CommonTypes
Annotations	OPTA string. Maximum length is 24 characters.
Diagram	
Type	restriction of xs:normalizedString
Facets	maxLength 24
Used by	Element ct:typeAddress/ct:opta
Source	<pre>&lt;xs:simpleType name="typeOPTA"&gt;   &lt;xs:annotation&gt;     &lt;xs:documentation&gt;OPTA string. Maximum length is 24 characters.&lt;/xs:documentation&gt;   &lt;/xs:annotation&gt;   &lt;xs:restriction base="xs:normalizedString"&gt;     &lt;xs:maxLength value="24"/&gt;   &lt;/xs:restriction&gt; &lt;/xs:simpleType&gt;</pre>

## Simple Type ct:typeAddressingStyle

Namespace	DR-GW-Interface/CommonTypes
Annotations	Describes the IP addressing style. Unicast or multicast.
Diagram	
Type	restriction of xs:normalizedString
Facets	enumeration ucast enumeration mcast
Source	<pre>&lt;xs:simpleType name="typeAddressingStyle"&gt;   &lt;xs:annotation&gt;     &lt;xs:documentation&gt;Describes the IP addressing style. Unicast or multicast.&lt;/xs:documentation&gt;   &lt;/xs:annotation&gt;   &lt;xs:restriction base="xs:normalizedString"&gt;     &lt;xs:enumeration value="ucast"/&gt;     &lt;xs:enumeration value="mcast"/&gt;   &lt;/xs:restriction&gt; &lt;/xs:simpleType&gt;</pre>