

# Schema documentation for DR-GW-Status.Events.xsd

november 5, 2024

## Table of Contents

Namespace: "DR-GW-Interface/DR-GW-Status.Events"	2
Schema(s) .....	2
Main schema DR-GW-Status.Events.xsd .....	2
Element(s) .....	2
Element Status_Response .....	2
Element Status_SendEvent .....	2
Element Status_SendEvent / status .....	3
Element Status_ReceiveEvent .....	3
Element Status_ReceiveEvent / status .....	4
Namespace: "DR-GW-Interface/DR-GW-Status.CommonTypes"	5
Schema(s) .....	5
Imported schema DR-GW-Status.CommonTypes.xsd .....	5
Element(s) .....	5
Element typeStatus / value .....	5
Element typeStatus / hexValue .....	5
Element typeStatus / source .....	5
Element typeStatus / target .....	6
Element typeStatus / tstamp .....	7
Complex Type(s) .....	7
Complex Type typeStatus .....	7
Namespace: "DR-GW-Interface/CommonTypes"	8
Schema(s) .....	8
Imported schema CommonTypes.xsd .....	8
Element(s) .....	8
Element ct:typeResponse / ct:requestId .....	8
Element ct:typeResponse / ct:result .....	8
Element ct:typeResult / ct:responseCode .....	9
Element ct:typeResult / ct:sourceSystem .....	9
Element ct:typeResult / ct:result .....	9
Element ct:typeEvent / ct:requestId .....	10
Element ct:typeEvent / ct:result .....	10
Element ct:typeAddress / ct:subscriber .....	10
Element ct:typeSubscriberAddress / ct:ssi .....	11
Element ct:typeSubscriberAddress / ct:tsi .....	11
Element ct:typeTSI / ct:mnc .....	12
Element ct:typeTSI / ct:mcc .....	12
Element ct:typeTSI / ct:ssi .....	12
Element ct:typeAddress / ct:alias .....	12
Element ct:typeAddress / ct:msisdn .....	12
Element ct:typeAddress / ct:fssn .....	13
Element ct:typeAddress / ct:external .....	13
Element ct:typeExternal / ct:gatewayNumber .....	13
Element ct:typeExternal / ct:number .....	14
Element ct:typeAddress / ct:opta .....	14
Element ct:typeAddress / ct:cell .....	14
Element ct:typeRequest / ct:requestId .....	14
Complex Type(s) .....	15
Complex Type ct:typeResponse .....	15
Complex Type ct:typeResult .....	15
Complex Type ct:typeEvent .....	15
Complex Type ct:typeAddress .....	16
Complex Type ct:typeSubscriberAddress .....	16
Complex Type ct:typeTSI .....	17
Complex Type ct:typeExternal .....	17
Complex Type ct:typeRequest .....	18
Complex Type ct:typeEmpty .....	18
Simple Type(s) .....	18
Simple Type ct:typeResponseCode .....	18
Simple Type ct:typeSourceSystem .....	18
Simple Type ct:typeDialString .....	19

Simple Type ct:typeOPTA .....	19
Simple Type ct:typeAddressingStyle .....	20

## Namespace: "DR-GW-Interface/DR-GW-Status.Events"

### Schema(s)

#### Main schema DR-GW-Status.Events.xsd

Namespace	DR-GW-Interface/DR-GW-Status.Events
Annotations	Version 1.1.1
Properties	attribute form default: unqualified element form default: qualified

### Element(s)

#### Element Status\_Response

Namespace	DR-GW-Interface/DR-GW-Status.Events
Annotations	
Diagram	<pre> classDiagram     ct:typeResponse &lt; -- Status_Response     Status_Response {         &lt;&lt;ct:typeResponse&gt;&gt;         requestId         result     }   </pre> <p>Response contains result of execution of any method.</p>
Type	ct:typeResponse
Properties	content: complex
Model	ct:requestId , ct:result
Children	ct:requestId, ct:result
Instance	<pre> &lt;Status_Response xmlns="DR-GW-Interface/DR-GW-Status.Events" xmlns:ct="DR-GW-Interface/CommonTypes"&gt;   &lt;ct:requestId&gt;{1,1}&lt;/ct:requestId&gt;   &lt;ct:result&gt;{1,1}&lt;/ct:result&gt; &lt;/Status_Response&gt;   </pre>
Source	<pre> &lt;x:element name="Status_Response" type="ct:typeResponse"&gt;   &lt;x:annotation&gt;     &lt;x:documentation/&gt;   &lt;/x:annotation&gt; &lt;/x:element&gt;   </pre>

#### Element Status\_SendEvent

Namespace	DR-GW-Interface/DR-GW-Status.Events
Annotations	
Diagram	<pre> classDiagram     ct:typeEvent &lt; -- Status_SendEvent     Status_SendEvent {         &lt;&lt;Extension of 'ct:typeEvent'&gt;&gt;         requestId         result         status     }   </pre>

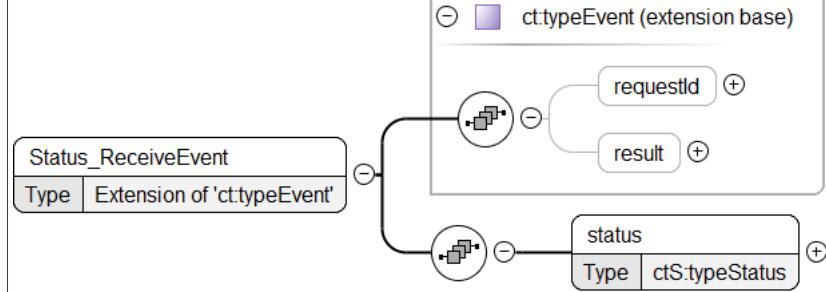
Type	extension of ct:typeEvent
Type hierarchy	• ct:typeEvent
Properties	content: complex
Model	ct:requestId{0,1} , ct:result{0,1} , status{0,1}
Children	ct:requestId, ct:result, status
Instance	<pre>&lt;Status_SendEvent xmlns="DR-GW-Interface/DR-GW-Status.Events" xmlns:ct="DR-GW-Interface/CommonTypes"&gt;   &lt;ct:requestId&gt;{0,1}&lt;/ct:requestId&gt;   &lt;ct:result&gt;{0,1}&lt;/ct:result&gt;   &lt;status&gt;{0,1}&lt;/status&gt; &lt;/Status_SendEvent&gt;</pre>
Source	<pre>&lt;xss:element name="Status_SendEvent"&gt;   &lt;xss:annotation&gt;     &lt;xss:documentation/&gt;   &lt;/xss:annotation&gt;   &lt;xss:complexType&gt;     &lt;xss:complexContent&gt;       &lt;xss:extension base="ct:typeEvent"&gt;         &lt;xss:sequence&gt;           &lt;xss:element name="status" type="cts:typeStatus" minOccurs="0"/&gt;         &lt;/xss:sequence&gt;       &lt;/xss:extension&gt;     &lt;/xss:complexContent&gt;   &lt;/xss:complexType&gt; &lt;/xss:element&gt;</pre>

## Element Status\_SendEvent / status

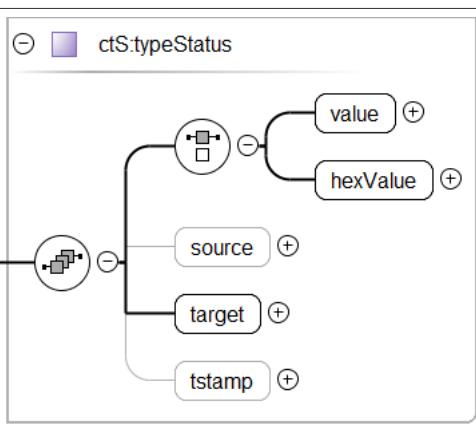
Namespace	DR-GW-Interface/DR-GW-Status.Events				
Diagram	<pre> classDiagram     class ctS:typeStatus {         value         hexValue         source         target         tstamp     }     class status {         &lt;&lt;Type ctS:typeStatus&gt;&gt;     }     status &lt; -- ctS:typeStatus   </pre>				
Type	typeStatus				
Properties	<table border="1"> <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	(value   hexValue) , source{0,1} , target , tstamp{0,1}				
Children	hexValue, source, target, tstamp, value				
Instance	<pre>&lt;status xmlns="DR-GW-Interface/DR-GW-Status.Events" xmlns:cts="DR-GW-Interface/DR-GW-Status.CommonTypes"&gt;   &lt;cts:value&gt;{1,1}&lt;/cts:value&gt;   &lt;cts:hexValue&gt;{1,1}&lt;/cts:hexValue&gt;   &lt;cts:source&gt;{0,1}&lt;/cts:source&gt;   &lt;cts:target&gt;{1,1}&lt;/cts:target&gt;   &lt;cts:tstamp&gt;{0,1}&lt;/cts:tstamp&gt; &lt;/status&gt;</pre>				
Source	<pre>&lt;xss:element name="status" type="cts:typeStatus" minOccurs="0"/&gt;</pre>				

## Element Status\_ReceiveEvent

Namespace	DR-GW-Interface/DR-GW-Status.Events
Annotations	

Diagram	
Type	extension of ct:typeEvent
Type hierarchy	<ul style="list-style-type: none"> <li>• ct:typeEvent</li> </ul>
Properties	content: complex
Model	ct:requestId{0,1} , ct:result{0,1} , status
Children	ct:requestId, ct:result, status
Instance	<pre>&lt;Status_ReceiveEvent xmlns="DR-GW-Interface/DR-GW-Status.Events" xmlns:ct="DR-GW-Interface/CommonTypes"&gt;   &lt;ct:requestId&gt;{0,1}&lt;/ct:requestId&gt;   &lt;ct:result&gt;{0,1}&lt;/ct:result&gt;   &lt;status&gt;{1,1}&lt;/status&gt; &lt;/Status_ReceiveEvent&gt;</pre>
Source	<pre>&lt;xss:element name="Status_ReceiveEvent"&gt;   &lt;xss:annotation&gt;     &lt;xss:documentation/&gt;   &lt;/xss:annotation&gt;   &lt;xss:complexType&gt;     &lt;xss:complexContent&gt;       &lt;xss:extension base="ct:typeEvent"&gt;         &lt;xss:sequence&gt;           &lt;xss:element name="status" type="cts:typeStatus" /&gt;         &lt;/xss:sequence&gt;       &lt;/xss:extension&gt;     &lt;/xss:complexContent&gt;   &lt;/xss:complexType&gt; &lt;/xss:element&gt;</pre>

## Element status\_ReceiveEvent / status

Namespace	DR-GW-Interface/DR-GW-Status.Events
Diagram	
Type	typeStatus
Properties	content: complex
Model	(value   hexValue) , source{0,1} , target , tstamp{0,1}
Children	hexValue, source, target, tstamp, value
Instance	<pre>&lt;status xmlns="DR-GW-Interface/DR-GW-Status.Events" xmlns:cts="DR-GW-Interface/DR-GW-Status.CommonTypes"&gt;   &lt;cts:value&gt;{1,1}&lt;/cts:value&gt;   &lt;cts:hexValue&gt;{1,1}&lt;/cts:hexValue&gt;   &lt;cts:source&gt;{0,1}&lt;/cts:source&gt;</pre>

	<pre>&lt;cts:target&gt;{1,1}&lt;/cts:target&gt; &lt;cts:tstamp&gt;{0,1}&lt;/cts:tstamp&gt; &lt;/status&gt;</pre>
Source	<code>&lt;xs:element name="status" type="cts:typeStatus" /&gt;</code>

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

### Schema(s)

Imported schema DR-GW-Status.CommonTypes.xsd

Namespace	DR-GW-Interface/DR-GW-Status.CommonTypes
Annotations	Version 1.1.1
Properties	attribute form default: unqualified element form default: qualified

### Element(s)

#### Element typeStatus / value

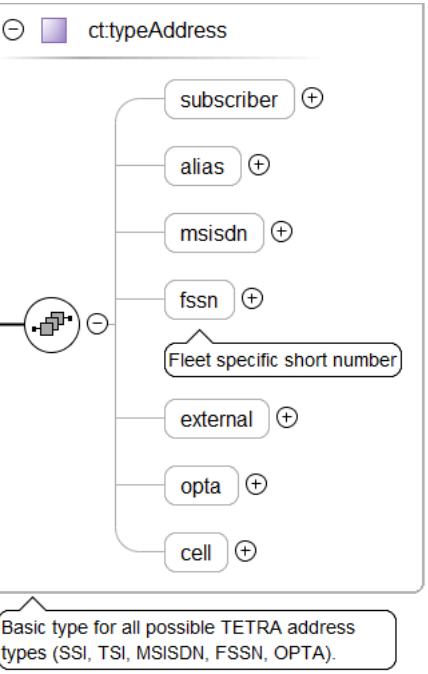
Namespace	DR-GW-Interface/DR-GW-Status.CommonTypes
Diagram	<p>The diagram shows a class named 'value' with a multiplicity of 0..1. It has a directed association labeled with a circled minus sign (⊖) pointing to a box containing 'xs:unsignedShort'. A callout box below the association states: 'Built-in derived type. The unsignedShort datatype is derived from unsignedInt by setting the value of maxInclusive to...'.</p>
Type	xs:unsignedShort
Properties	content: simple
Source	<code>&lt;xs:element name="value" type="xs:unsignedShort" /&gt;</code>

#### Element typeStatus / hexValue

Namespace	DR-GW-Interface/DR-GW-Status.CommonTypes
Diagram	<p>The diagram shows a class named 'hexValue' with a multiplicity of 0..1. It has a directed association labeled with a circled minus sign (⊖) pointing to a box containing 'xs:hexBinary'. A callout box below the association states: 'Built-in primitive type. The hexBinary datatype represents arbitrary hex-encoded binary data.'</p>
Type	xs:hexBinary
Properties	content: simple
Source	<code>&lt;xs:element name="hexValue" type="xs:hexBinary" /&gt;</code>

#### Element typeStatus / source

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

Diagram					
Type	ct:typeAddress				
Properties	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="padding: 2px;">content:</td><td style="padding: 2px;">complex</td></tr> <tr> <td style="padding: 2px;">minOccurs:</td><td style="padding: 2px;">0</td></tr> </table>	content:	complex	minOccurs:	0
content:	complex				
minOccurs:	0				
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				
Instance	<pre>&lt;source xmlns="DR-GW-Interface/DR-GW-Status.CommonTypes" xmlns:ct="DR-GW-Interface/CommonTypes"&gt;   &lt;ct:subscriber&gt;{0,1}&lt;/ct:subscriber&gt;   &lt;ct:alias&gt;{0,1}&lt;/ct:alias&gt;   &lt;ct:msisdn&gt;{0,1}&lt;/ct:msisdn&gt;   &lt;ct:fssn&gt;{0,1}&lt;/ct:fssn&gt;   &lt;ct:external&gt;{0,1}&lt;/ct:external&gt;   &lt;ct:opta&gt;{0,1}&lt;/ct:opta&gt;   &lt;ct:cell&gt;{0,1}&lt;/ct:cell&gt; &lt;/source&gt;</pre>				
Source	<pre>&lt;xss:element name="source" type="ct:typeAddress" minOccurs="0"/&gt;</pre>				

## Element typestatus / target

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

Diagram	<p>Basic type for all possible TETRA address types (SSI, TSI, MSISDN, FSSN, OPTA).</p>
Type	ct:typeAddress
Properties	content: complex
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
Instance	<pre>&lt;target xmlns="DR-GW-Interface/DR-GW-Status.CommonTypes" xmlns:ct="DR-GW-Interface/CommonTypes"&gt;   &lt;ct:subscriber&gt;{0,1}&lt;/ct:subscriber&gt;   &lt;ct:alias&gt;{0,1}&lt;/ct:alias&gt;   &lt;ct:msisdn&gt;{0,1}&lt;/ct:msisdn&gt;   &lt;ct:fssn&gt;{0,1}&lt;/ct:fssn&gt;   &lt;ct:external&gt;{0,1}&lt;/ct:external&gt;   &lt;ct:opta&gt;{0,1}&lt;/ct:opta&gt;   &lt;ct:cell&gt;{0,1}&lt;/ct:cell&gt; &lt;/target&gt;</pre>
Source	<code>&lt;xss:element name="target" type="ct:typeAddress" /&gt;</code>

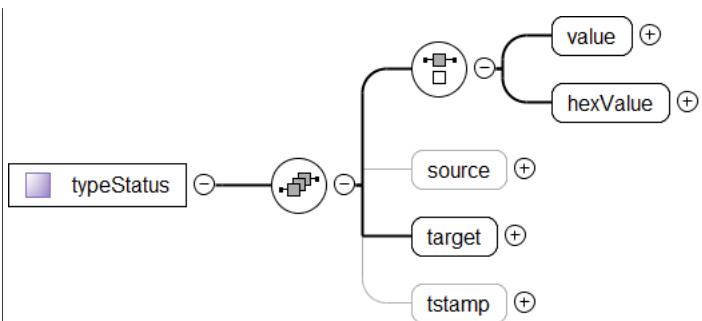
### Element typeStatus / tstamp

Namespace	DR-GW-Interface/DR-GW-Status.CommonTypes				
Diagram					
Type	xs:dateTime				
Properties	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="padding: 2px;">content:</td> <td style="padding: 2px;">simple</td> </tr> <tr> <td style="padding: 2px;">minOccurs:</td> <td style="padding: 2px;">0</td> </tr> </table>	content:	simple	minOccurs:	0
content:	simple				
minOccurs:	0				
Source	<code>&lt;xss:element name="tstamp" type="xs:dateTime" minOccurs="0" /&gt;</code>				

## Complex Type(s)

### Complex Type typeStatus

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

Diagram	
Used by	Elements      Status_ReceiveEvent/status, Status_SendEvent/status
Model	(value   hexValue) , source{0,1} , target , tstamp{0,1}
Children	hexValue, source, target, tstamp, value
Source	<pre>&lt;xs:complexType name="typeStatus"&gt;   &lt;xs:sequence&gt;     &lt;xs:choice&gt;       &lt;xs:element name="value" type="xs:unsignedShort"/&gt;       &lt;xs:element name="hexValue" type="xs:hexBinary"/&gt;     &lt;/xs:choice&gt;     &lt;xs:element name="source" type="ct:typeAddress" minOccurs="0"/&gt;     &lt;xs:element name="target" type="ct:typeAddress"/&gt;     &lt;xs:element name="tstamp" type="xs:dateTime" minOccurs="0"/&gt;   &lt;/xs:sequence&gt; &lt;/xs:complexType&gt;</pre>

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

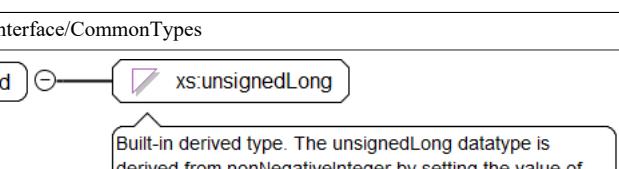
### Schema(s)

Imported schema CommonTypes.xsd

Namespace	DR-GW-Interface/CommonTypes
Annotations	Version 1.1.1
Properties	attribute form default: unqualified element form default: qualified

### Element(s)

Element ct:typeResponse / ct:requestId

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

Element ct:typeResponse / ct:result

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

Diagram	<pre> classDiagram     class ct:typeResult {         result *--&gt; responseCode         result *--&gt; sourceSystem         result *--&gt; result     }     note over ct:typeResult: Common result values used in every response and optional specific subsystem result codes.   </pre>
Type	ct:typeResult
Properties	content: complex
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	<xss:element name="result" type="ct:typeResult"/>

### Element ct:typeResult / ct:responseCode

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

### Element ct:typeResult / ct:sourceSystem

Namespace	DR-GW-Interface/CommonTypes						
Diagram	<pre> classDiagram     class SourceSystem     class ct:typeSourceSystem     SourceSystem *--&gt; ct:typeSourceSystem   </pre>						
Type	ct:typeSourceSystem						
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>enumeration</td><td>DR-GW</td></tr> <tr><td>enumeration</td><td>TCS-API</td></tr> <tr><td>enumeration</td><td>TETRA</td></tr> </table>	enumeration	DR-GW	enumeration	TCS-API	enumeration	TETRA
enumeration	DR-GW						
enumeration	TCS-API						
enumeration	TETRA						
Source	<xss:element name="sourceSystem" type="ct:typeSourceSystem" minOccurs="0"/>						

### Element ct:typeResult / ct:result

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

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	<code>&lt;xs:element name="result" type="xs:unsignedLong" minOccurs="0" /&gt;</code>				

### Element ct:typeEvent / ct:requestId

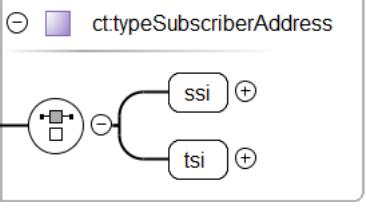
Namespace	DR-GW-Interface/CommonTypes				
Diagram					
Type	xs:unsignedLong				
Properties	<table> <tr> <td>content:</td><td>simple</td></tr> <tr> <td>minOccurs:</td><td>0</td></tr> </table>	content:	simple	minOccurs:	0
content:	simple				
minOccurs:	0				
Source	<code>&lt;xs:element name="requestId"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

### Element ct:typeEvent / ct:result

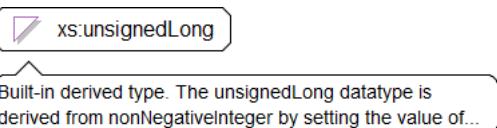
Namespace	DR-GW-Interface/CommonTypes				
Diagram					
Type	ct:typeResult				
Properties	<table> <tr> <td>content:</td><td>complex</td></tr> <tr> <td>minOccurs:</td><td>0</td></tr> </table>	content:	complex	minOccurs:	0
content:	complex				
minOccurs:	0				
Model	ct:responseCode , ct:sourceSystem{0,1}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

### Element ct:typeAddress / ct:subscriber

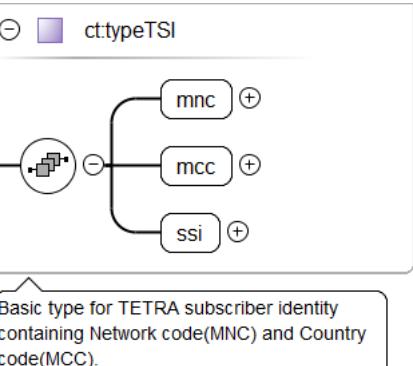
Namespace	DR-GW-Interface/CommonTypes
-----------	-----------------------------

Diagram					
Type	ct:typeSubscriberAddress				
Properties	<table border="1"> <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: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	<code>&lt;xss:element name="subscriber" type="ct:typeSubscriberAddress" minOccurs="0"/&gt;</code>				

### Element ct:typeSubscriberAddress / ct:ssi

Namespace	DR-GW-Interface/CommonTypes		
Diagram			
Type	xs:unsignedLong		
Properties	<table border="1"> <tr> <td>content:</td> <td>simple</td> </tr> </table>	content:	simple
content:	simple		
Source	<code>&lt;xss:element name="ssi" type="xs:unsignedLong"/&gt;</code>		

### Element ct:typeSubscriberAddress / ct:tsi

Namespace	DR-GW-Interface/CommonTypes		
Diagram			
Type	ct:typeTSI		
Properties	<table border="1"> <tr> <td>content:</td> <td>complex</td> </tr> </table>	content:	complex
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;xss:element name="tsi" type="ct:typeTSI"/&gt;</code>		

### Element ct:typeTSI / ct:mnc

Namespace	DR-GW-Interface/CommonTypes
Diagram	<p>Built-in derived type. The unsignedShort datatype is derived from unsignedInt by setting the value of maxInclusive to...</p>
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	<p>Built-in derived type. The unsignedShort datatype is derived from unsignedInt by setting the value of maxInclusive to...</p>
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	<p>Built-in derived type. The unsignedLong datatype is derived from nonNegativeInteger by setting the value of...</p>
Type	xs:unsignedLong
Properties	content: simple
Source	<code>&lt;xs:element name="ssi" type="xs:unsignedLong" /&gt;</code>

### Element ct:typeAddress / ct:alias

Namespace	DR-GW-Interface/CommonTypes
Diagram	<p>Built-in derived type. The normalizedString datatype represents white space normalized strings. The base type of...</p>
Type	xs:normalizedString
Properties	content: simple minOccurs: 0
Source	<code>&lt;xs:element name="alias" type="xs:normalizedString" minOccurs="0" /&gt;</code>

### Element ct:typeAddress / ct:msisdn

Namespace	DR-GW-Interface/CommonTypes
Diagram	<p>Allowed characters are digits 0 - 9, *, #, A, B, C and D. Maximum length is 24 characters.</p>

Type	ct:typeDialString
Properties	content: simple minOccurs: 0
Facets	maxLength 24
Source	<code>&lt;xs:element name="msisdn" type="ct:typeDialString" minOccurs="0" /&gt;</code>

### Element ct:typeAddress / ct:fssn

Namespace	DR-GW-Interface/CommonTypes
Annotations	Fleet specific short number
Diagram	<pre> classDiagram     class fssn {         &lt;&lt;Fleet specific short number&gt;&gt;     }     class xs_unsignedLong {         &lt;&lt;Built-in derived type. The unsignedLong datatype is derived from nonNegativeInteger by setting the value of...&gt;&gt;     }     fssn &lt; -- xs_unsignedLong   </pre>
Type	xs:unsignedLong
Properties	content: simple minOccurs: 0
Source	<code>&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;</code>

### Element ct:typeAddress / ct:external

Namespace	DR-GW-Interface/CommonTypes
Diagram	<pre> classDiagram     class external {         &lt;&lt;External number consisting of Gateway number + DialString&gt;&gt;     }     class ctypeExternal {         &lt;&lt;cctypeExternal&gt;&gt;     }     class gatewayNumber     class number     external &lt; -- ctypeExternal     ctypeExternal &lt; -- gatewayNumber     ctypeExternal &lt; -- number   </pre>
Type	ct:typeExternal
Properties	content: complex minOccurs: 0
Model	ct:gatewayNumber , ct:number
Children	ct:gatewayNumber, ct:number
Instance	<code>&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;</code>
Source	<code>&lt;xs:element name="external" type="ct:typeExternal" minOccurs="0" /&gt;</code>

### Element ct:typeExternal / ct:gatewayNumber

Namespace	DR-GW-Interface/CommonTypes
Diagram	<pre> classDiagram     class gatewayNumber {         &lt;&lt;Built-in derived type. The unsignedLong datatype is derived from nonNegativeInteger by setting the value of...&gt;&gt;     }     class xs_unsignedLong {         &lt;&lt;xs:unsignedLong&gt;&gt;     }     gatewayNumber &lt; -- xs_unsignedLong   </pre>
Type	xs:unsignedLong

Properties	content: simple
Source	<xs:element name="gatewayNumber" type="xs:unsignedLong" />

### Element ct:typeExternal / ct:number

Namespace	DR-GW-Interface/CommonTypes
Diagram	
Type	ct:typeDialString
Properties	content: simple
Facets	maxLength 24
Source	<xs:element name="number" type="ct:typeDialString" />

### Element ct:typeAddress / ct:opta

Namespace	DR-GW-Interface/CommonTypes
Diagram	
Type	ct:typeOPTA
Properties	content: simple minOccurs: 0
Facets	maxLength 24
Source	<xs:element name="opta" type="ct:typeOPTA" minOccurs="0" />

### Element ct:typeAddress / ct:cell

Namespace	DR-GW-Interface/CommonTypes
Diagram	
Type	xs:short
Properties	content: simple minOccurs: 0
Source	<xs:element name="cell" type="xs:short" minOccurs="0" />

### Element ct:typeRequest / ct:requestId

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

## Complex Type(s)

### Complex Type ct:typeResponse

Namespace	DR-GW-Interface/CommonTypes
Annotations	Response contains result of execution of any method.
Diagram	<pre> classDiagram     class typeResponse     class requestId     class result      typeResponse "1" -- "1" requestId :      typeResponse "1" -- "1" result :    </pre> <p>Response contains result of execution of any method.</p>
Used by	Element Status_Response
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:typeResult

Namespace	DR-GW-Interface/CommonTypes
Annotations	Common result values used in every response and optional specific subsystem result codes.
Diagram	<pre> classDiagram     class typeResult     class responseCode     class sourceSystem     class result      typeResult "1" -- "1" responseCode :      typeResult "1" -- "1" sourceSystem :      typeResult "1" -- "1" result :    </pre> <p>Common result values used in every response and optional specific subsystem result codes.</p>
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:typeEvent

Namespace	DR-GW-Interface/CommonTypes
Diagram	<pre> classDiagram     class typeEvent     class requestId     class result      typeEvent "1" -- "1" requestId :      typeEvent "1" -- "1" result :    </pre>

Used by	Elements	Status_ReceiveEvent, Status_SendEvent
Model	ct:requestId{0,1} , ct:result{0,1}	
Children	ct:requestId, ct:result	
Source	<pre>&lt;xss:complexType name="typeEvent"&gt;   &lt;xss:sequence&gt;     &lt;xss:element name="requestId" type="xs:unsignedLong" minOccurs="0"/&gt;     &lt;xss:element name="result" type="ct:typeResult" minOccurs="0"/&gt;   &lt;/xss:sequence&gt; &lt;/xss: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	<pre> classDiagram     class typeAddress {         &lt;&lt;Basic type for all possible TETRA address types (SSI, TSI, MSISDN, FSSN, OPTA).&gt;&gt;     }     class subscriber     class alias     class msisdn     class fssn     class external     class opta     class cell      typeAddress "0..1" -- "0..1" subscriber : +     typeAddress "0..1" -- "0..1" alias : +     typeAddress "0..1" -- "0..1" msisdn : +     typeAddress "0..1" -- "0..1" fssn : +     typeAddress "0..1" -- "0..1" external : +     typeAddress "0..1" -- "0..1" opta : +     typeAddress "0..1" -- "0..1" cell : +   </pre>
Used by	Elements
	typeStatus/source, typeStatus/target
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;xss:complexType name="typeAddress"&gt;   &lt;xss:annotation&gt;     &lt;xss:documentation&gt;Basic type for all possible TETRA address types (SSI, TSI, MSISDN, FSSN, OPTA).&lt;/xss:documentation&gt;   &lt;/xss:annotation&gt;   &lt;xss:sequence&gt;     &lt;xss:element name="subscriber" type="ct:typeSubscriberAddress" minOccurs="0"/&gt;     &lt;xss:element name="alias" type="xs:normalizedString" minOccurs="0"/&gt;     &lt;xss:element name="msisdn" type="ct:typeDialString" minOccurs="0"/&gt;     &lt;xss:element name="fssn" type="xs:unsignedLong" minOccurs="0"&gt;       &lt;xss:annotation&gt;         &lt;xss:documentation&gt;Fleet specific short number&lt;/xss:documentation&gt;       &lt;/xss:annotation&gt;     &lt;/xss:element&gt;     &lt;xss:element name="external" type="ct:typeExternal" minOccurs="0"/&gt;     &lt;xss:element name="opta" type="ct:typeOPTA" minOccurs="0"/&gt;     &lt;xss:element name="cell" type="xs:short" minOccurs="0"/&gt;   &lt;/xss:sequence&gt; &lt;/xss:complexType&gt;</pre>

## Complex Type ct:typeSubscriberAddress

Namespace	DR-GW-Interface/CommonTypes
Annotations	
Diagram	<pre> classDiagram     class typeSubscriberAddress {     }     class ssi     class tsi      typeSubscriberAddress "0..1" -- "0..1" ssi : +     typeSubscriberAddress "0..1" -- "0..1" tsi : +   </pre>
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>

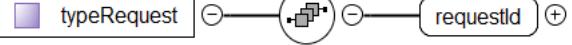
### Complex Type ct:typeTSI

Namespace	DR-GW-Interface/CommonTypes
Annotations	Basic type for TETRA subscriber identity containing Network code(MNC) and Country code(MCC).
Diagram	<pre> classDiagram     class typeTSI {         mnc         mcc         ssi     }     mnc "*" --&gt; "2" typeTSI     mcc "*" --&gt; "2" typeTSI     ssi "*" --&gt; "2" typeTSI   </pre> <p>Basic type for TETRA subscriber identity containing Network code(MNC) and Country code(MCC).</p>
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	<pre> classDiagram     class typeExternal {         gatewayNumber         number     }     gatewayNumber "*" --&gt; "2" typeExternal     number "*" --&gt; "2" typeExternal   </pre> <p>External number consisting of Gateway number + DialString</p>
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:typeRequest

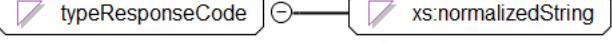
Namespace	DR-GW-Interface/CommonTypes
Diagram	
Model	ct:requestId
Children	ct:requestId
Source	<pre>&lt;xs:complexType name="typeRequest"&gt;   &lt;xs:sequence&gt;     &lt;xs:element name="requestId" type="xs:unsignedLong" /&gt;   &lt;/xs:sequence&gt; &lt;/xs:complexType&gt;</pre>

## Complex Type ct:typeEmpty

Namespace	DR-GW-Interface/CommonTypes
Annotations	Explicit type specification for elements that shall be empty.
Diagram	 A callout box points from the diagram to the annotation text: "Explicit type specification for elements that shall be empty."
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	 A callout box points from the diagram to the annotation text: "Built-in derived type. The normalizedString datatype represents white space normalized strings. The base type of..."												
Type	restriction of xs:normalizedString												
Facets	<table> <tr> <td>enumeration</td> <td>success</td> </tr> <tr> <td>enumeration</td> <td>final_response_pending</td> </tr> <tr> <td>enumeration</td> <td>error</td> </tr> <tr> <td>enumeration</td> <td>not_authorized_error</td> </tr> <tr> <td>enumeration</td> <td>temporary_failure</td> </tr> <tr> <td>enumeration</td> <td>subscription_failed</td> </tr> </table>	enumeration	success	enumeration	final_response_pending	enumeration	error	enumeration	not_authorized_error	enumeration	temporary_failure	enumeration	subscription_failed
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 bases="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	<table> <tr> <td>enumeration</td> <td>DR-GW</td> </tr> <tr> <td>enumeration</td> <td>TCS-API</td> </tr> <tr> <td>enumeration</td> <td>TETRA</td> </tr> </table>	enumeration	DR-GW	enumeration	TCS-API	enumeration	TETRA
enumeration	DR-GW						
enumeration	TCS-API						
enumeration	TETRA						
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: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>

### 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;</pre>

```

<xs:maxLength value="24" />
</xs:restriction>
</xs:simpleType>

```

## Simple Type ct:typeAddressingStyle

Namespace	DR-GW-Interface/CommonTypes				
Annotations	Describes the IP addressing style. Unicast or multicast.				
Diagram	<pre> classDiagram     typeAddressingStyle &lt; -- xs:normalizedString     </pre> <p>The diagram shows two rounded rectangles representing UML classes. The left one is labeled "typeAddressingStyle" and the right one is labeled "xs:normalizedString". A directed association line connects them, with an open circle at the "typeAddressingStyle" end and a filled circle at the "xs:normalizedString" end. Below the diagram are two callout boxes. The left one points to "typeAddressingStyle" and contains the text "Describes the IP addressing style. Unicast or multicast.". The right one points to "xs:normalizedString" and contains the text "Built-in derived type. The normalizedString datatype represents white space normalized strings. The base type of...".   </p>				
Type	restriction of xs:normalizedString				
Facets	<table> <tr> <td>enumeration</td> <td>ucast</td> </tr> <tr> <td>enumeration</td> <td>mcast</td> </tr> </table>	enumeration	ucast	enumeration	mcast
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>				