

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

november 5, 2024

## Table of Contents

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

Simple Type ct:typeDialString .....	18
Simple Type ct:typeOPTA .....	18
Simple Type ct:typeAddressingStyle .....	19

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

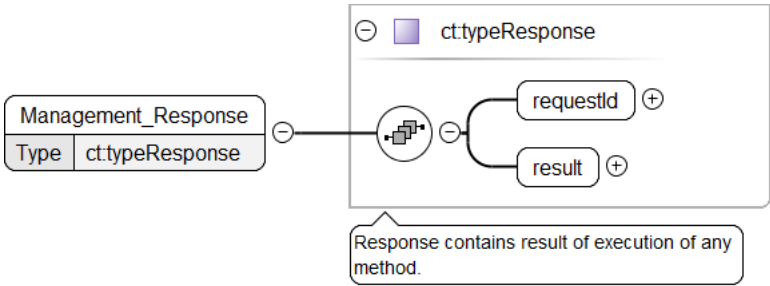
### Schema(s)

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

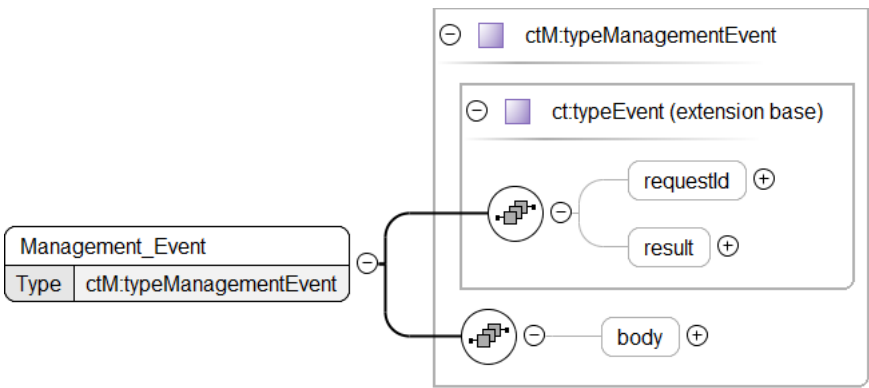
Namespace	DR-GW-Interface/DR-GW-Management.Events
Annotations	Version 1.2  DR-GW-Management. Use for so called NEM-API(Tactilon-API) functions. Use only via SOAP.
Properties	attribute form default: unqualified element form default: qualified

### Element(s)

#### Element Management\_Response

Namespace	DR-GW-Interface/DR-GW-Management.Events
Diagram	
Type	ct:typeResponse
Properties	content: complex
Model	ct:requestId, ct:result
Children	ct:requestId, ct:result
Instance	<pre>&lt;Management_Response xmlns="DR-GW-Interface/DR-GW-Management.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;/Management_Response&gt;</pre>
Source	<pre>&lt;xs:element name="Management_Response" type="ct:typeResponse" /&gt;</pre>

#### Element Management\_Event

Namespace	DR-GW-Interface/DR-GW-Management.Events
Diagram	

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

## 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 typeManagementEvent / body

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

#### Element typeManagementRequest / requestUri

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

#### 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	<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="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	<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	username , password				
Children	password, username				
Instance	<pre>&lt;authentication xmlns="DR-GW-Interface/DR-GW-Management.CommonTypes"&gt;   &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	<code>&lt;xs:element name="username" type="xs:string" /&gt;</code>

## Element typeManagementAuthentication / password

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

## 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>

## Complex Type(s)

### 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>
Used by	Element Management_Event
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;       &lt;xs:sequence&gt;         &lt;xs:element name="body" type="xs:string" minOccurs="0"/&gt;       &lt;/xs:sequence&gt;     &lt;/xs:extension&gt;   &lt;/xs:complexContent&gt; &lt;/xs:complexType&gt;</pre>

## Complex Type typeManagementRequest

Namespace	DR-GW-Interface/DR-GW-Management.CommonTypes
Annotations	Generic management request used for all GET, POST, PUT, DELETE operations.
Diagram	
Type	extension of ct:typeRequest
Type hierarchy	<ul style="list-style-type: none"> <li>ct:typeRequest</li> <li>typeManagementRequest</li> </ul>
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: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:annotation&gt;</pre>

```

<xs:documentation>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.</
xs:documentation>
</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>

## 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: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>

## Element `ct:typeResult` / `ct:responseCode`

Namespace	DR-GW-Interface/CommonTypes												
Diagram													
Type	<code>ct:typeResponseCode</code>												
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	<code>&lt;xs:element name="responseCode" type="ct:typeResponseCode" /&gt;</code>												

## Element `ct:typeResult` / `ct:sourceSystem`

Namespace	DR-GW-Interface/CommonTypes				
Diagram					
Type	<code>ct:typeSourceSystem</code>				
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> </table>	enumeration	DR-GW	enumeration	TCS-API
enumeration	DR-GW				
enumeration	TCS-API				



	enumeration	TETRA
	enumeration	TACTILON-API
Source	<code>&lt;xs:element name="sourceSystem" type="ct:typeSourceSystem" minOccurs="0"/&gt;</code>	

## Element ct:typeResult / ct:result

Namespace	DR-GW-Interface/CommonTypes	
Diagram		
Type	xs:unsignedLong	
Properties	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	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	<code>&lt;xs:element name="result" type="ct:typeResult" minOccurs="0"/&gt;</code>	

### 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: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`

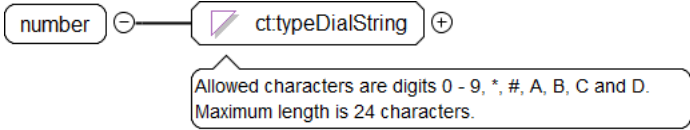
Namespace	DR-GW-Interface/CommonTypes
Diagram	
Type	xs:unsignedLong
Properties	content: simple
Source	<code>&lt;xs:element name="ssi" type="xs:unsignedLong"/&gt;</code>

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

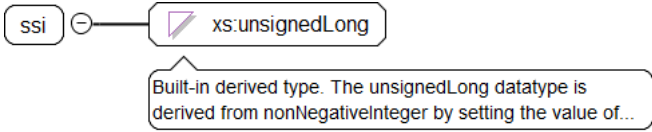
Namespace	DR-GW-Interface/CommonTypes
Diagram	
Type	xs:unsignedLong

Properties	content: simple
Source	<a href="#">&lt;xs:element name="gatewayNumber" type="xs:unsignedLong" /&gt;</a>

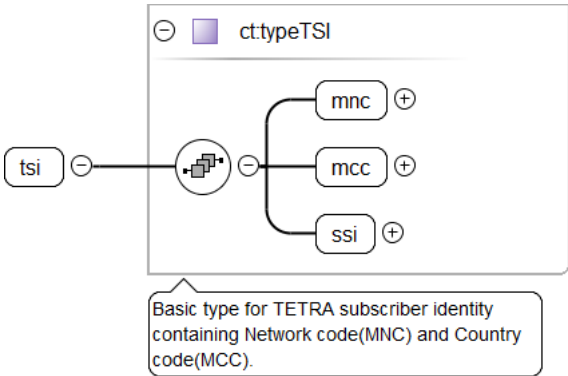
## Element `ct:typeExternal` / `ct:number`

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

## Element `ct:typeSubscriberAddress` / `ct:ssi`

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

## 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	<a href="#">&lt;xs:element name="tsi" type="ct:typeTSI" /&gt;</a>

## 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	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	content: simple minOccurs: 0
Facets	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	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	<code>ct:typeExternal</code>
Properties	content: complex
	minOccurs: 0
Model	<code>ct:gatewayNumber</code> , <code>ct:number</code>
Children	<code>ct:gatewayNumber</code> , <code>ct:number</code>
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	<code>ct:typeOPTA</code>
Properties	content: simple
	minOccurs: 0
Facets	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	<code>xs:short</code>
Properties	content: simple
	minOccurs: 0
Source	<pre>&lt;xs:element name="cell" type="xs:short" minOccurs="0"/&gt;</pre>

## Complex Type(s)

### Complex Type `ct:typeResponse`

Namespace	DR-GW-Interface/CommonTypes
Annotations	Response contains result of execution of any method.
Diagram	
Used by	Element <code>Management_Response</code>
Model	<code>ct:requestId</code> , <code>ct:result</code>
Children	<code>ct:requestId</code> , <code>ct:result</code>
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	
Used by	Elements <code>ct:typeEvent/ct:result</code> , <code>ct:typeResponse/ct:result</code>
Model	<code>ct:responseCode</code> , <code>ct:sourceSystem{0,1}</code> , <code>ct:result{0,1}</code>
Children	<code>ct:responseCode</code> , <code>ct:result</code> , <code>ct:sourceSystem</code>
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	
Used by	Complex Type <code>typeManagementEvent</code>

Model	ct:requestId{0,1} , ct:result{0,1}
Children	ct:requestId, ct:result
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:element name="result" type="ct:typeResult" minOccurs="0"/&gt;   &lt;/xs:sequence&gt; &lt;/xs:complexType&gt; </pre>

## Complex Type ct:typeRequest

Namespace	DR-GW-Interface/CommonTypes
Diagram	
Used by	Complex Type typeManagementRequest
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: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 consiting 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; </pre>

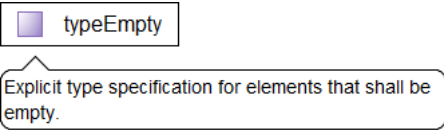


```

</xs:annotation>
<xs:sequence>
  <xs:element name="subscriber" type="ct:typeSubscriberAddress" minOccurs="0"/>
  <xs:element name="alias" type="xs:normalizedString" minOccurs="0"/>
  <xs:element name="msisdn" type="ct:typeDialString" minOccurs="0"/>
  <xs:element name="fssn" type="xs:unsignedLong" minOccurs="0">
    <xs:annotation>
      <xs:documentation>Fleet specific short number</xs:documentation>
    </xs:annotation>
  </xs:element>
  <xs:element name="external" type="ct:typeExternal" minOccurs="0"/>
  <xs:element name="opta" type="ct:typeOPTA" minOccurs="0"/>
  <xs:element name="cell" type="xs:short" minOccurs="0"/>
</xs:sequence>
</xs:complexType>

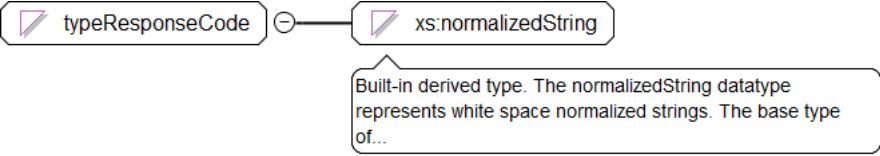
```

## 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;/   &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	<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 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	<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> <tr> <td>enumeration</td><td>TACTILON-API</td></tr> </table>	enumeration	DR-GW	enumeration	TCS-API	enumeration	TETRA	enumeration	TACTILON-API
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>

### 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>

```
</xs:annotation>
<xs:restriction base="xs:normalizedString">
  <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> graph LR     typeAddressingStyle -- restriction --&gt; xs_normalizedString[xs:normalizedString]     typeAddressingStyle --- callout1[Describes the IP addressing style. Unicast or multicast.]     xs_normalizedString --- callout2[Built-in derived type. The normalizedString datatype represents white space normalized strings. The base type of...]   </pre>				
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>				