

Schema documentation for DR-GW-Session.Events.xsd

november 5, 2024

Table of Contents

Namespace: "DR-GW-Interface/DR-GW-Session.Events"	1
Schema(s)	1
Main schema DR-GW-Session.Events.xsd	1
Element(s)	2
Element Session_Response	2
Element Session_LoginEvent	2
Element Session_LoginEvent / issi	3
Element Session_LogoutEvent	3
Element Session_LogoutEvent / reason	4
Element Session_SuperviseEvent	4
Namespace: "DR-GW-Interface/CommonTypes"	4
Schema(s)	4
Imported schema CommonTypes.xsd	4
Element(s)	5
Element ct:typeResponse / ct:requestId	5
Element ct:typeResponse / ct:result	5
Element ct:typeResult / ct:responseCode	5
Element ct:typeResult / ct:sourceSystem	6
Element ct:typeResult / ct:result	6
Element ct:typeEvent / ct:requestId	6
Element ct:typeEvent / ct:result	6
Element ct:typeTSI / ct:mnc	7
Element ct:typeTSI / ct:mcc	7
Element ct:typeTSI / ct:ssi	7
Element ct:typeExternal / ct:gatewayNumber	7
Element ct:typeExternal / ct:number	8
Element ct:typeSubscriberAddress / ct:ssi	8
Element ct:typeSubscriberAddress / ct:tsi	8
Element ct:typeAddress / ct:subscriber	8
Element ct:typeAddress / ct:alias	9
Element ct:typeAddress / ct:msisdn	9
Element ct:typeAddress / ct:fssn	9
Element ct:typeAddress / ct:external	10
Element ct:typeAddress / ct:opta	10
Element ct:typeAddress / ct:cell	10
Element ct:typeRequest / ct:requestId	11
Complex Type(s)	11
Complex Type ct:typeResponse	11
Complex Type ct:typeResult	11
Complex Type ct:typeEvent	12
Complex Type ct:typeTSI	12
Complex Type ct:typeExternal	12
Complex Type ct:typeSubscriberAddress	13
Complex Type ct:typeAddress	13
Complex Type ct:typeRequest	14
Complex Type ct:typeEmpty	14
Simple Type(s)	14
Simple Type ct:typeResponseCode	14
Simple Type ct:typeSourceSystem	15
Simple Type ct:typeDialString	15
Simple Type ct:typeOPTA	15
Simple Type ct:typeAddressingStyle	16

Namespace: "DR-GW-Interface/DR-GW-Session.Events"

Schema(s)

Main schema DR-GW-Session.Events.xsd

Namespace	DR-GW-Interface/DR-GW-Session.Events
-----------	--------------------------------------

Annotations	Version 1.1.1
Properties	attribute form default: unqualified
	element form default: qualified

Element(s)

Element Session_Response

Namespace	DR-GW-Interface/DR-GW-Session.Events
Annotations	
Diagram	<pre> classDiagram class Session_Response { <<ct:typeResponse>> } Session_Response < -- ct:typeResponse ct:typeResponse < -- ct:requestId ct:typeResponse < -- ct:result ct:requestId --> requestId ct:result --> result note over ct:typeResponse: Response contains result of execution of any method. </pre>
Type	ct:typeResponse
Properties	content: complex
Model	ct:requestId , ct:result
Children	ct:requestId, ct:result
Instance	<pre> <Session_Response xmlns="DR-GW-Interface/DR-GW-Session.Events" xmlns:ct="DR-GW-Interface/CommonTypes"> <ct:requestId>{1,1}</ct:requestId> <ct:result>{1,1}</ct:result> </Session_Response> </pre>
Source	<pre> <xss:element name="Session_Response" type="ct:typeResponse"> <xss:annotation> <xss:documentation/> </xss:annotation> </xss:element> </pre>

Element Session_LoginEvent

Namespace	DR-GW-Interface/DR-GW-Session.Events
Annotations	
Diagram	<pre> classDiagram class Session_LoginEvent { <<Extension of 'ct:typeEvent'>> } Session_LoginEvent < -- ct:typeEvent ct:typeEvent < -- ct:requestId ct:typeEvent < -- ct:result ct:typeEvent < -- issi issi < -- xs:string ct:requestId --> requestId ct:result --> result issi --> issi </pre>
Type	extension of ct:typeEvent
Type hierarchy	<ul style="list-style-type: none"> • ct:typeEvent
Properties	content: complex
Model	ct:requestId{0,1} , ct:result{0,1} , issi{0,1}
Children	ct:requestId, ct:result, issi
Instance	<pre> <Session_LoginEvent xmlns="DR-GW-Interface/DR-GW-Session.Events" xmlns:ct="DR-GW-Interface/CommonTypes"> <ct:requestId>{0,1}</ct:requestId> <ct:result>{0,1}</ct:result> </Session_LoginEvent> </pre>

	<pre><issi>{0,1}</issi> </Session_LoginEvent></pre>
Source	<pre><x:element name="Session_LoginEvent"> <x:annotation> <x:documentation/> </x:annotation> <x:complexType> <x:complexContent> <x:extension base="ct:typeEvent"> <x:sequence> <x:element name="issi" type="xs:string" minOccurs="0"/> </x:sequence> </x:extension> </x:complexContent> </x:complexType> </x:element></pre>

Element Session_LoginEvent / issi

Namespace	DR-GW-Interface/DR-GW-Session.Events				
Diagram	<p>The diagram shows the 'issi' element as a simple type xs:string. It consists of a rounded rectangle labeled 'issi' with a small minus sign (-) to its left, indicating it's a simple type. To its right is a rounded rectangle labeled 'xs:string' with a small plus sign (+) to its right, indicating it's the base type. A line connects them.</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	<pre><x:element name="issi" type="xs:string" minOccurs="0"/></pre>				

Element Session_LogoutEvent

Namespace	DR-GW-Interface/DR-GW-Session.Events
Annotations	
Diagram	<p>The diagram shows the 'Session_LogoutEvent' element as an extension of 'ct:typeEvent'. It consists of a rounded rectangle labeled 'Session_LogoutEvent' with a small minus sign (-) to its left, indicating it's an extension. To its right is a rounded rectangle labeled 'Extension of 'ct:typeEvent'' with a small plus sign (+) to its right, indicating it's the base type. A line connects them. Inside the 'Session_LogoutEvent' box, there are three child elements: 'requestId' (with a minus sign), 'result' (with a plus sign), and 'reason' (with a plus sign). Each child element has a small icon to its left.</p>
Type	extension of ct:typeEvent
Type hierarchy	<ul style="list-style-type: none"> • ct:typeEvent
Properties	content: complex
Model	ct:requestId{0,1} , ct:result{0,1} , reason{0,1}
Children	ct:requestId, ct:result, reason
Instance	<pre><Session_LogoutEvent xmlns="DR-GW-Interface/DR-GW-Session.Events" xmlns:ct="DR-GW-Interface/CommonTypes"> <ct:requestId>{0,1}</ct:requestId> <ct:result>{0,1}</ct:result> <reason>{0,1}</reason> </Session_LogoutEvent></pre>
Source	<pre><x:element name="Session_LogoutEvent"> <x:annotation> <x:documentation/> </x:annotation> <x:complexType> <x:complexContent> <x:extension base="ct:typeEvent"> <x:sequence></pre>

```

        <xs:element name="reason" type="xs:unsignedLong" minOccurs="0" />
    </xs:sequence>
</xs:extension>
</xs:complexContent>
</xs:complexType>
</xs:element>

```

Element Session_LogoutEvent / reason

Namespace	DR-GW-Interface/DR-GW-Session.Events				
Diagram	<p>The diagram shows the 'reason' element as a class with a 'Type' association pointing to the 'xs:unsignedLong' class. A callout box indicates that 'xs:unsignedLong' is a built-in derived type derived from 'nonNegativeInteger'.</p>				
Type	xs:unsignedLong				
Properties	<table border="1"> <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	<xs:element name="reason" type="xs:unsignedLong" minOccurs="0" />				

Element Session_SuperviseEvent

Namespace	DR-GW-Interface/DR-GW-Session.Events
Annotations	
Diagram	<p>The diagram shows 'Session_SuperviseEvent' as a class extending 'ct:typeEvent'. It has two attributes: 'requestId' and 'result'.</p>
Type	extension of ct:typeEvent
Type hierarchy	• ct:typeEvent
Properties	content: complex
Model	ct:requestId{0,1} , ct:result{0,1}
Children	ct:requestId, ct:result
Instance	<Session_SuperviseEvent xmlns="DR-GW-Interface/DR-GW-Session.Events" xmlns:ct="DR-GW-Interface/CommonTypes"> <ct:requestId>{0,1}</ct:requestId> <ct:result>{0,1}</ct:result> </Session_SuperviseEvent>
Source	<xs:element name="Session_SuperviseEvent"> <xs:annotation> <xs:documentation/> </xs:annotation> <xs:complexType> <xs:complexContent> <xs:extension base="ct:typeEvent"> <xs:extension> </xs:extension> </xs:complexContent> </xs:complexType> </xs:element>

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	<p>The diagram shows a class named 'requestId' with a multiplicity of 0..1. It has a directed association to a class represented by a purple icon (xs:unsignedLong) with a multiplicity of 1..1. A callout box indicates that 'xs:unsignedLong' is a built-in derived type derived from nonNegativeInteger.</p>
Type	xs:unsignedLong
Properties	content: simple
Source	<code><xss:element name="requestId" type="xs:unsignedLong"/></code>

Element ct:typeResponse / ct:result

Namespace	DR-GW-Interface/CommonTypes
Diagram	<p>The diagram shows a class named 'result' with a multiplicity of 0..1. It has three associations: one to 'responseCode' (multiplicity 1..1), one to 'sourceSystem' (multiplicity 0..1), and one to 'result' (multiplicity 0..1). A callout box states that 'result' is a common result value used in every response and optional specific subsystem result codes.</p>
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> <ct:result xmlns:ct="DR-GW-Interface/CommonTypes"> <ct:responseCode>{1,1}</ct:responseCode> <ct:sourceSystem>{0,1}</ct:sourceSystem> <ct:result>{0,1}</ct:result> </ct:result> </pre>
Source	<code><xss:element name="result" type="ct:typeResult"/></code>

Element ct:typeResult / ct:responseCode

Namespace	DR-GW-Interface/CommonTypes												
Diagram	<p>The diagram shows a class named 'responseCode' with a multiplicity of 0..1. It has a directed association to a class named 'ct:typeResponseCode' with a multiplicity of 1..1. A callout box indicates that 'ct:typeResponseCode' is a derived type.</p>												
Type	ct:typeResponseCode												
Properties	content: simple												
Facets	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="padding: 2px;">enumeration</td> <td style="padding: 2px;">success</td> </tr> <tr> <td style="padding: 2px;">enumeration</td> <td style="padding: 2px;">final_response_pending</td> </tr> <tr> <td style="padding: 2px;">enumeration</td> <td style="padding: 2px;">error</td> </tr> <tr> <td style="padding: 2px;">enumeration</td> <td style="padding: 2px;">not_authorized_error</td> </tr> <tr> <td style="padding: 2px;">enumeration</td> <td style="padding: 2px;">temporary_failure</td> </tr> <tr> <td style="padding: 2px;">enumeration</td> <td style="padding: 2px;">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><xss:element name="responseCode" type="ct:typeResponseCode" /></code>
--------	---

Element ct:typeResult / ct:sourceSystem

Namespace	DR-GW-Interface/CommonTypes						
Diagram	<pre> classDiagram class sourceSystem class ct:typeSourceSystem sourceSystem "0..1" -- "0..1" ct:typeSourceSystem </pre>						
Type	ct:typeSourceSystem						
Properties	<table border="1"> <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 border="1"> <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	<code><xss:element name="sourceSystem" type="ct:typeSourceSystem" minOccurs="0" /></code>						

Element ct:typeResult / ct:result

Namespace	DR-GW-Interface/CommonTypes				
Diagram	<pre> classDiagram class result class xs:unsignedLong result "0..1" -- "0..1" xs:unsignedLong </pre> <p>Built-in derived type. The unsignedLong datatype is derived from nonNegativeInteger by setting the value of...</p>				
Type	xs:unsignedLong				
Properties	<table border="1"> <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><xss:element name="result" type="xs:unsignedLong" minOccurs="0" /></code>				

Element ct:typeEvent / ct:requestId

Namespace	DR-GW-Interface/CommonTypes				
Diagram	<pre> classDiagram class requestId class xs:unsignedLong requestId "0..1" -- "0..1" xs:unsignedLong </pre> <p>Built-in derived type. The unsignedLong datatype is derived from nonNegativeInteger by setting the value of...</p>				
Type	xs:unsignedLong				
Properties	<table border="1"> <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><xss:element name="requestId" type="xs:unsignedLong" minOccurs="0" /></code>				

Element ct:typeEvent / ct:result

Namespace	DR-GW-Interface/CommonTypes
Diagram	<pre> classDiagram class result class ct:typeResult class responseCode class sourceSystem class result result "0..1" -- "0..1" ct:typeResult ct:typeResult "*" -- "0..1" responseCode ct:typeResult "*" -- "0..1" sourceSystem ct:typeResult "*" -- "0..1" result </pre> <p>Common result values used in every response and optional specific subsystem result codes.</p>

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><ct:result xmlns:ct="DR-GW-Interface/CommonTypes"> <ct:responseCode>{1,1}</ct:responseCode> <ct:sourceSystem>{0,1}</ct:sourceSystem> <ct:result>{0,1}</ct:result> </ct:result></pre>
Source	<code><xss:element name="result" type="ct:typeResult" minOccurs="0" /></code>

Element ct:typeTSI / ct:mnc

Namespace	DR-GW-Interface/CommonTypes
Diagram	<p>The diagram shows a class named 'mnc' with a dependency arrow pointing to 'xs:unsignedShort'. A callout box indicates that 'mnc' is a built-in derived type derived from 'xs:unsignedShort'.</p>
Type	xs:unsignedShort
Properties	content: simple
Source	<code><xss:element name="mnc" type="xs:unsignedShort" /></code>

Element ct:typeTSI / ct:mcc

Namespace	DR-GW-Interface/CommonTypes
Diagram	<p>The diagram shows a class named 'mcc' with a dependency arrow pointing to 'xs:unsignedShort'. A callout box indicates that 'mcc' is a built-in derived type derived from 'xs:unsignedShort'.</p>
Type	xs:unsignedShort
Properties	content: simple
Source	<code><xss:element name="mcc" type="xs:unsignedShort" /></code>

Element ct:typeTSI / ct:ssi

Namespace	DR-GW-Interface/CommonTypes
Diagram	<p>The diagram shows a class named 'ssi' with a dependency arrow pointing to 'xs:unsignedLong'. A callout box indicates that 'ssi' is a built-in derived type derived from 'xs:unsignedLong'.</p>
Type	xs:unsignedLong
Properties	content: simple
Source	<code><xss:element name="ssi" type="xs:unsignedLong" /></code>

Element ct:typeExternal / ct:gatewayNumber

Namespace	DR-GW-Interface/CommonTypes
Diagram	<p>The diagram shows a class named 'gatewayNumber' with a dependency arrow pointing to 'xs:unsignedLong'. A callout box indicates that 'gatewayNumber' is a built-in derived type derived from 'xs:unsignedLong'.</p>
Type	xs:unsignedLong

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

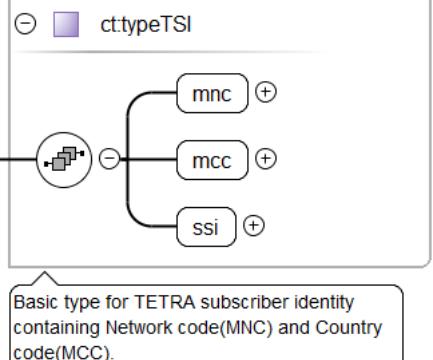
Element ct:typeExternal / ct:number

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:typeSubscriberAddress / ct:ssi

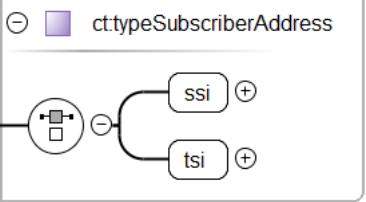
Namespace	DR-GW-Interface/CommonTypes
Diagram	
Type	xs:unsignedLong
Properties	content: simple
Source	<xs:element name="ssi" type="xs:unsignedLong" />

Element ct:typeSubscriberAddress / ct:tsi

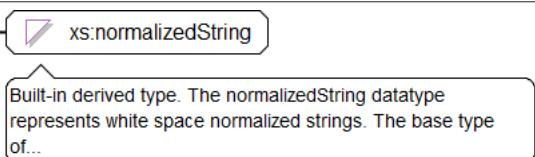
Namespace	DR-GW-Interface/CommonTypes
Diagram	
Type	ct:typeTSI
Properties	content: complex
Model	ct:mnc , ct:mcc , ct:ssi
Children	ct:mcc, ct:mnc, ct:ssi
Instance	<pre> <ct:tsi xmlns:ct="DR-GW-Interface/CommonTypes"> <ct:mnc>{1,1}</ct:mnc> <ct:mcc>{1,1}</ct:mcc> <ct:ssi>{1,1}</ct:ssi> </ct:tsi> </pre>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

Element ct:typeAddress / ct:subscriber

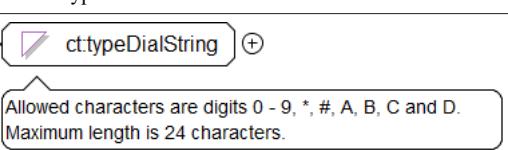
Namespace	DR-GW-Interface/CommonTypes
-----------	-----------------------------

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><ct:subscriber xmlns:ct="DR-GW-Interface/CommonTypes"> <ct:ssi>{1,1}</ct:ssi> <ct:tsi>{1,1}</ct:tsi> </ct:subscriber></pre>				
Source	<code><xss:element name="subscriber" type="ct:typeSubscriberAddress" minOccurs="0" /></code>				

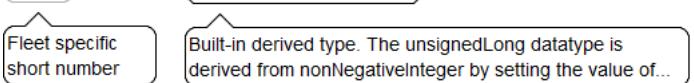
Element ct:typeAddress / ct:alias

Namespace	DR-GW-Interface/CommonTypes				
Diagram					
Type	xs:normalizedString				
Properties	<table border="1"> <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><xss:element name="alias" type="xs:normalizedString" minOccurs="0" /></code>				

Element ct:typeAddress / ct:msisdn

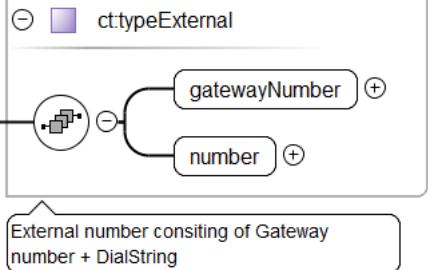
Namespace	DR-GW-Interface/CommonTypes				
Diagram					
Type	ct:typeDialString				
Properties	<table border="1"> <tr> <td>content:</td> <td>simple</td> </tr> <tr> <td>minOccurs:</td> <td>0</td> </tr> </table>	content:	simple	minOccurs:	0
content:	simple				
minOccurs:	0				
Facets	maxLength 24				
Source	<code><xss:element name="msisdn" type="ct:typeDialString" minOccurs="0" /></code>				

Element ct:typeAddress / ct:fssn

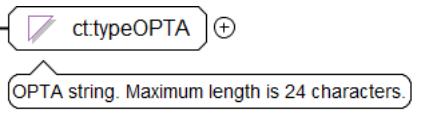
Namespace	DR-GW-Interface/CommonTypes
Annotations	Fleet specific short number
Diagram	
Type	xs:unsignedLong

Properties	content: simple minOccurs: 0
Source	<pre><xs:element name="fssn" type="xs:unsignedLong" minOccurs="0"> <xs:annotation> <xs:documentation>Fleet specific short number</xs:documentation> </xs:annotation> </xs:element></pre>

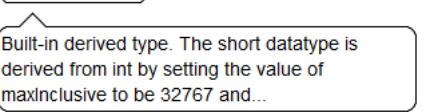
Element ct:typeAddress / ct:external

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

Element ct:typeAddress / ct:opta

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

Element ct:typeAddress / ct:cell

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

Element ct:typeRequest / ct:requestId

Namespace	DR-GW-Interface/CommonTypes
Diagram	<p>The diagram shows the element 'requestId' connected to the datatype 'xs:unsignedLong'. A callout box states: 'Built-in derived type. The unsignedLong datatype is derived from nonNegativeInteger by setting the value of...'.</p>
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	<p>The diagram shows the element 'typeResponse' with two children: 'requestId' and 'result', both marked with a plus sign indicating they are optional. A callout box states: 'Response contains result of execution of any method.'</p>
Used by	Element Session_Response
Model	ct:requestId , ct:result
Children	ct:requestId, ct:result
Source	<pre><xs:complexType name="typeResponse"> <xs:annotation> <xs:documentation>Response contains result of execution of any method.</xs:documentation> </xs:annotation> <xs:sequence> <xs:element name="requestId" type="xs:unsignedLong"/> <xs:element name="result" type="ct:typeResult"/> </xs:sequence> </xs:complexType></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	<p>The diagram shows the element 'typeResult' with three children: 'responseCode', 'sourceSystem', and 'result', all marked with a plus sign indicating they are optional. A callout box states: '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><xs:complexType name="typeResult"> <xs:annotation> <xs:documentation>Common result values used in every response and optional specific subsystem result codes.</xs:documentation> </xs:annotation> <xs:sequence> <xs:element name="responseCode" type="ct:typeResponseCode"/></pre>

```

<xs:element name="sourceSystem" type="ct:typeSourceSystem" minOccurs="0" />
<xs:element name="result" type="xs:unsignedLong" minOccurs="0" />
</xs:sequence>
</xs:complexType>

```

Complex Type ct:typeEvent

Namespace	DR-GW-Interface/CommonTypes
Diagram	
Used by	Elements Session_LoginEvent, Session_LogoutEvent, Session_SuperviseEvent
Model	ct:requestId{0,1} , ct:result{0,1}
Children	ct:requestId, ct:result
Source	<pre> <xs:complexType name="typeEvent"> <xs:sequence> <xs:element name="requestId" type="xs:unsignedLong" minOccurs="0" /> <xs:element name="result" type="ct:typeResult" minOccurs="0" /> </xs:sequence> </xs:complexType> </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> <xs:complexType name="typeTSI"> <xs:annotation> <xs:documentation>Basic type for TETRA subscriber identity containing Network code(MNC) and Country code(MCC).</xs:documentation> </xs:annotation> <xs:sequence> <xs:element name="mnc" type="xs:unsignedShort" /> <xs:element name="mcc" type="xs:unsignedShort" /> <xs:element name="ssi" type="xs:unsignedLong" /> </xs:sequence> </xs:complexType> </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><xs:complexType name="typeExternal"> <xs:annotation> <xs:documentation>External number consisting of Gateway number + DialString</xs:documentation> </xs:annotation> <xs:sequence> <xs:element name="gatewayNumber" type="xs:unsignedLong"/> <xs:element name="number" type="ct:typeDialString"/> </xs:sequence> </xs:complexType></pre>

Complex Type ct:typeSubscriberAddress

Namespace	DR-GW-Interface/CommonTypes
Annotations	
Diagram	<pre> classDiagram class typeSubscriberAddress { <<choice>> <<ssi : unsignedLong>> <<tsi : typeTSI>> } </pre>
Used by	Element ct:typeAddress/ct:subscriber
Model	ct:ssi ct:tsi
Children	ct:ssi, ct:tsi
Source	<pre><xs:complexType name="typeSubscriberAddress"> <xs:annotation> <xs:documentation/> </xs:annotation> <xs:choice> <xs:element name="ssi" type="xs:unsignedLong"/> <xs:element name="tsi" type="ct:typeTSI"/> </xs:choice> </xs:complexType></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 { <<choice>> <<subscriber : typeSubscriberAddress>> <<alias : normalizedString>> <<msisdn : string>> <<fssn : string>> <<external : string>> <<opta : string>> <<cell : string>> } </pre> <p>Basic type for all possible TETRA address types (SSI, TSI, MSISDN, FSSN, OPTA).</p>
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><xs:complexType name="typeAddress"> <xs:annotation> <xs:documentation>Basic type for all possible TETRA address types (SSI, TSI, MSISDN, FSSN, OPTA).</xs:documentation> </xs:annotation> <xs:sequence> <xs:element name="subscriber" type="ct:typeSubscriberAddress" minOccurs="0"/> <xs:element name="alias" type="xs:normalizedString" minOccurs="0"/> </xs:sequence> </xs:complexType></pre>

```

<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:typeRequest

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

Complex Type ct:typeEmpty

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

Simple Type(s)

Simple Type ct:typeResponseCode

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	restriction of xs:normalizedString												
Facets	<table border="1"> <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> <xs:simpleType name="typeResponseCode"> <xs:restriction base="xs:normalizedString"> <xs:enumeration value="success"/> </xs:restriction> </xs:simpleType> </pre>												

```

<xs:enumeration value="final_response_pending" />
<xs:enumeration value="error" />
<xs:enumeration value="not_authorized_error" />
<xs:enumeration value="temporary_failure" />
<xs:enumeration value="subscription_failed" />
</xs:restriction>
</xs:simpleType>

```

Simple Type ct:typeSourceSystem

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	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> <xs:simpleType name="typeSourceSystem"> <xs:restriction bases="xs:normalizedString"> <xs:enumeration value="DR-GW" /> <xs:enumeration value="TCS-API" /> <xs:enumeration value="TETRA" /> </xs:restriction> </xs:simpleType> </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	<p>Allowed characters are digits 0 - 9, *, #, A, B, C and D. Maximum length is 24 characters.</p> <p>Built-in derived type. The normalizedString datatype represents white space normalized strings. The base type of...</p>
Type	restriction of xs:normalizedString
Facets	maxLength 24
Used by	Elements ct:typeAddress/ct:msisdn, ct:typeExternal/ct:number
Source	<pre> <xs:simpleType name="typeDialString"> <xs:annotation> <xs:documentation>Allowed characters are digits 0 - 9, *, #, A, B, C and D. Maximum length is 24 characters.</xs:documentation> </xs:annotation> <xs:restriction base="xs:normalizedString"> <xs:maxLength value="24" /> </xs:restriction> </xs:simpleType> </pre>

Simple Type ct:typeOPTA

Namespace	DR-GW-Interface/CommonTypes
Annotations	OPTA string. Maximum length is 24 characters.
Diagram	<p>OPTA string. Maximum length is 24 characters.</p> <p>Built-in derived type. The normalizedString datatype represents white space normalized strings. The base type of...</p>

Type	restriction of xs:normalizedString	
Facets	maxLength	24
Used by	Element	ct:typeAddress/ct:opta
Source	<pre><xs:simpleType name="typeOPTA"> <xs:annotation> <xs:documentation>OPTA string. Maximum length is 24 characters.</xs:documentation> </xs:annotation> <xs:restriction base="xs:normalizedString"> <xs:maxLength value="24"/> </xs:restriction> </xs:simpleType></pre>	

Simple Type ct:typeAddressingStyle

Namespace	DR-GW-Interface/CommonTypes				
Annotations	Describes the IP addressing style. Unicast or multicast.				
Diagram	<pre> classDiagram class typeAddressingStyle { <<Describes the IP addressing style. Unicast or multicast.>> } class xs_normalizedString { <<Built-in derived type. The normalizedString datatype represents white space normalized strings. The base type of...>> } typeAddressingStyle < -- xs_normalizedString </pre>				
Type	restriction of xs:normalizedString				
Facets	<table border="1"> <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><xs:simpleType name="typeAddressingStyle"> <xs:annotation> <xs:documentation>Describes the IP addressing style. Unicast or multicast.</xs:documentation> </xs:annotation> <xs:restriction base="xs:normalizedString"> <xs:enumeration value="ucast"/> <xs:enumeration value="mcast"/> </xs:restriction> </xs:simpleType></pre>				