

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

november 5, 2024

Table of Contents

Namespace: "DR-GW-Interface/DR-GW-Radio.Events"	3
Schema(s)	3
Main schema DR-GW-Radio.Events.xsd	3
Element(s)	3
Element Radio_Response	3
Element Radio_GetEvent	3
Element Radio_GetEvent / radio	4
Element Radio_GetListEvent	4
Element Radio_GetListEvent / radio	5
Element Radio_GetListEvent / listEnd	6
Element Radio_GetGroupsEvent	6
Element Radio_GetGroupsEvent / radio	7
Element Radio_GetGroupsEvent / group	7
Element Radio_GetGroupsEvent / listEnd	7
Element Radio_GroupsEvent	8
Element Radio_GroupsEvent / radio	8
Element Radio_GroupsEvent / group	9
Element Radio_GroupsEvent / deletedGroup	9
Element Radio_Event	9
Element Radio_Event / radio	10
Element Radio_Event / delete	11
Element Radio_TrackEvent	11
Element Radio_TrackEvent / trackingData	11
Element Radio_ChangeOPTAEVENT	12
Element Radio_ChangeOPTAEVENT / radio	13
Element Radio_ChangeOPTAEVENT / opta	14
Element Radio_EnDisableEvent	14
Element Radio_EnDisableEvent / radio	15
Element Radio_EnDisableEvent / reason	15
Element Radio_EnDisableEvent / enabled	15
Element Radio_EnDisableEvent / overTheAir	15
Element Radio_TrackSubscriptionEvent	16
Element Radio_TrackSubscriptionEvent / radio	16
Element Radio_TrackSubscriptionEvent / stop	17
Namespace: "DR-GW-Interface/CommonTypes"	17
Schema(s)	17
Imported schema CommonTypes.xsd	17
Element(s)	17
Element ct:typeResponse / ct:requestId	17
Element ct:typeResponse / ct:result	17
Element ct:typeResult / ct:responseCode	18
Element ct:typeResult / ct:sourceSystem	18
Element ct:typeResult / ct:result	18
Element ct:typeEvent / ct:requestId	19
Element ct:typeEvent / ct:result	19
Element ct:typeSubscriberAddress / ct:ssi	19
Element ct:typeSubscriberAddress / ct:tsi	20
Element ct:typeTSI / ct:mnc	20
Element ct:typeTSI / ct:mcc	20
Element ct:typeTSI / ct:ssi	20
Element ct:typeExternal / ct:gatewayNumber	21
Element ct:typeExternal / ct:number	21
Element ct:typeAddress / ct:subscriber	21
Element ct:typeAddress / ct:alias	21
Element ct:typeAddress / ct:msisdn	22
Element ct:typeAddress / ct:fssn	22
Element ct:typeAddress / ct:external	22
Element ct:typeAddress / ct:opta	23
Element ct:typeAddress / ct:cell	23
Element ct:typeRequest / ct:requestId	23

Complex Type(s)	23
Complex Type ct:typeResponse	23
Complex Type ct:typeResult	24
Complex Type ct:typeEvent	24
Complex Type ct:typeSubscriberAddress	25
Complex Type ct:typeTSI	25
Complex Type ct:typeExternal	25
Complex Type ct:typeAddress	26
Complex Type ct:typeRequest	26
Complex Type ct:typeEmpty	27
Simple Type(s)	27
Simple Type ct:typeResponseCode	27
Simple Type ct:typeSourceSystem	27
Simple Type ct:typeOPTA	28
Simple Type ct:typeDialString	28
Simple Type ct:typeAddressingStyle	29
Namespace: "DR-GW-Interface/DR-GW-Radio.CommonTypes"	29
Schema(s)	29
Imported schema DR-GW-Radio.CommonTypes.xsd	29
Element(s)	29
Element typeRadio / issi	29
Element typeRadio / alias	30
Element typeRadio / orgblockId	30
Element typeRadio / opta	30
Element typeLastKnownOPTA / tstamp	30
Element typeLastKnownOPTA / opta	31
Element typeRadioGroupSelection / group	31
Element typeRadioGroupSelection / level	31
Element typeRadioTrackingData / radio	32
Element typeRadioTrackingData / registered	32
Element typeRadioTrackingData / exchangeId	32
Element typeRadioTrackingData / locationArea	32
Element typeRadioTrackingData / lastActive	33
Element typeRadioTrackingData / scanningOn	33
Element typeRadioTrackingData / status	33
Element typeStatusIndicator / value	33
Element typeStatusIndicator / time	34
Element typeRadioTrackingData / callType	34
Element typeRadioTrackingData / callParty	34
Element typeRadioTrackingData / dmoState	34
Element typeRadioTrackingData / emergency	35
Complex Type(s)	35
Complex Type typeRadio	35
Complex Type typeLastKnownOPTA	35
Complex Type typeRadioGroupSelection	36
Complex Type typeRadioTrackingData	36
Complex Type typeStatusIndicator	37
Simple Type(s)	37
Simple Type typeGroupSelectionLevel	37
Simple Type typeCallType	38
Namespace: "DR-GW-Interface/DR-GW-OrganisationBlock.CommonTypes"	38
Schema(s)	38
Imported schema DR-GW-OrganisationBlock.CommonTypes.xsd	38
Element(s)	38
Element typeOrganisationBlockId / orgblockId	38
Element typeOrganisationBlockIdNormal / id1	39
Element typeOrganisationBlockIdNormal / id2	39
Element typeOrganisationBlockIdNormal / id3	39
Element typeOrganisationBlockIdNormal / id4	39
Element typeOrganisationBlockIdNormal / id5	40
Element typeOrganisationBlockIdNormal / id6	40
Element typeOrganisationBlockId / orgblockIdSimple	40
Element typeOrganisationBlock / orgblockId	41
Element typeOrganisationBlock / alias	41
Complex Type(s)	41
Complex Type typeOrganisationBlockId	41
Complex Type typeOrganisationBlockIdNormal	41
Complex Type typeOrganisationBlock	42
Simple Type(s)	42
Simple Type typeOrganisationBlockIdSimple	42

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

Schema(s)

Main schema DR-GW-Radio.Events.xsd

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

Element(s)

Element Radio_Response

Namespace	DR-GW-Interface/DR-GW-Radio.Events
Annotations	
Diagram	<pre> classDiagram ct:typeResponse < -- Radio_Response Radio_Response { requestId result } note over Radio_Response: 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> <Radio_Response xmlns="DR-GW-Interface/DR-GW-Radio.Events" xmlns:ct="DR-GW-Interface/CommonTypes"> <ct:requestId>{1,1}</ct:requestId> <ct:result>{1,1}</ct:result> </Radio_Response> </pre>
Source	<pre> <xsd:element name="Radio_Response" type="ct:typeResponse"> <xsd:annotation> <xsd:documentation/> </xsd:annotation> </xsd:element> </pre>

Element Radio_GetEvent

Namespace	DR-GW-Interface/DR-GW-Radio.Events
Annotations	
Diagram	<pre> ct:typeEvent < -- Radio_GetEvent Radio_GetEvent { requestId result } Radio_GetEvent --> ct:typeRadio : radio </pre>
Type	extension of ct:typeEvent
Type hierarchy	<ul style="list-style-type: none"> • ct:typeEvent

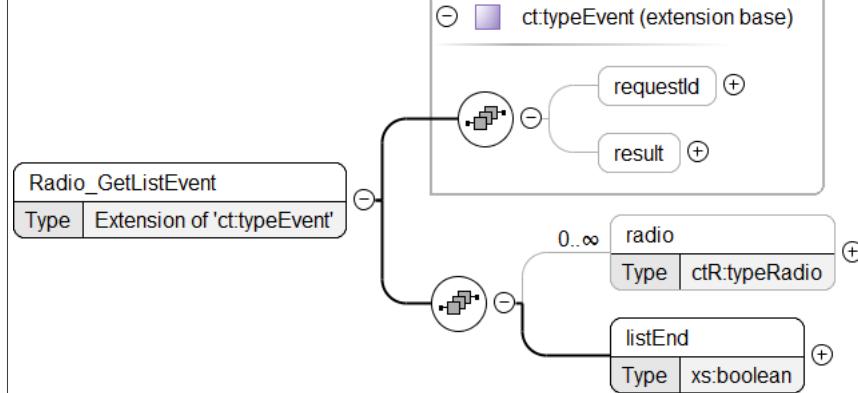
Properties	content: complex
Model	ct:requestId{0,1} , ct:result{0,1} , radio
Children	ct:requestId, ct:result, radio
Instance	<pre><Radio_GetEvent xmlns="DR-GW-Interface/DR-GW-Radio.Events" xmlns:ct="DR-GW-Interface/CommonTypes"> <ct:requestId>{0,1}</ct:requestId> <ct:result>{0,1}</ct:result> <radio>{1,1}</radio> </Radio_GetEvent></pre>
Source	<pre><xs:element name="Radio_GetEvent"> <xs:annotation> <xs:documentation/> </xs:annotation> <xs:complexType> <xs:complexContent> <xs:extension base="ct:typeEvent"> <xs:sequence> <xs:element name="radio" type="ctR:typeRadio"/> </xs:sequence> </xs:extension> </xs:complexContent> </xs:complexType> </xs:element></pre>

Element Radio_GetEvent / radio

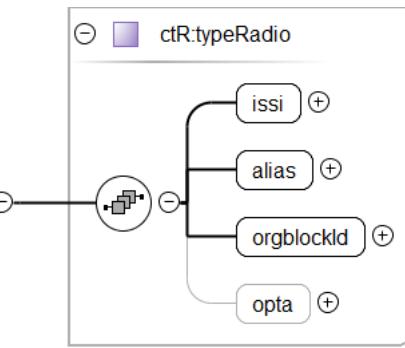
Namespace	DR-GW-Interface/DR-GW-Radio.Events
Diagram	
Type	typeRadio
Properties	content: complex
Model	issi , alias , orgblockId , opta{0,1}
Children	alias, issi, opta, orgblockId
Instance	<pre><radio xmlns="DR-GW-Interface/DR-GW-Radio.Events" xmlns:ctR="DR-GW-Interface/DR-GW-Radio.CommonTypes"> <ctR:issi>{1,1}</ctR:issi> <ctR:alias>{1,1}</ctR:alias> <ctR:orgblockId>{1,1}</ctR:orgblockId> <ctR:opta>{0,1}</ctR:opta> </radio></pre>
Source	<pre><xs:element name="radio" type="ctR:typeRadio"/></pre>

Element Radio_GetListEvent

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

Diagram	
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} , radio* , listEnd
Children	ct:requestId, ct:result, listEnd, radio
Instance	<pre><Radio_GetListEvent xmlns="DR-GW-Interface/DR-GW-Radio.Events" xmlns:ct="DR-GW-Interface/CommonTypes"> <ct:requestId>{0,1}</ct:requestId> <ct:result>{0,1}</ct:result> <radio>{0,unbounded}</radio> <listEnd>{1,1}</listEnd> </Radio_GetListEvent></pre>
Source	<pre><xs:element name="Radio_GetListEvent"> <xs:annotation> <xs:documentation/> </xs:annotation> <xs:complexType> <xs:complexContent> <xs:extension base="ct:typeEvent"> <xs:sequence> <xs:element name="radio" type="ctR:typeRadio" minOccurs="0" maxOccurs="unbounded" /> <xs:element name="listEnd" type="xs:boolean"/> </xs:sequence> </xs:extension> </xs:complexContent> </xs:complexType> </xs:element></pre>

Element Radio_GetListEvent / radio

Namespace	DR-GW-Interface/DR-GW-Radio.Events
Diagram	
Type	typeRadio
Properties	<p>content: complex</p> <p>minOccurs: 0</p> <p>maxOccurs: unbounded</p>
Model	issi , alias , orgblockId , opta{0,1}

Children	alias, issi, opta, orgblockId
Instance	<pre><radio xmlns="DR-GW-Interface/DR-GW-Radio.Events" xmlns:ctR="DR-GW-Interface/DR-GW-Radio.CommonTypes"> <ctR:issi>{1,1}</ctR:issi> <ctR:alias>{1,1}</ctR:alias> <ctR:orgblockId>{1,1}</ctR:orgblockId> <ctR:opta>{0,1}</ctR:opta> </radio></pre>
Source	<code><xss:element name="radio" type="ctR:typeRadio" minOccurs="0" maxOccurs="unbounded" /></code>

Element Radio_GetListEvent / listEnd

Namespace	DR-GW-Interface/DR-GW-Radio.Events
Diagram	
Type	xs:boolean
Properties	content: simple
Source	<code><xss:element name="listEnd" type="xs:boolean" /></code>

Element Radio_GetGroupsEvent

Namespace	DR-GW-Interface/DR-GW-Radio.Events
Annotations	
Diagram	
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} , radio , group* , listEnd
Children	ct:requestId, ct:result, group, listEnd, radio
Instance	<pre><Radio_GetGroupsEvent xmlns="DR-GW-Interface/DR-GW-Radio.Events" xmlns:ct="DR-GW-Interface/CommonTypes"> <ct:requestId>{0,1}</ct:requestId> <ct:result>{0,1}</ct:result> <radio>{1,1}</radio> <group>{0,unbounded}</group> <listEnd>{1,1}</listEnd> </Radio_GetGroupsEvent></pre>
Source	<pre><xss:element name="Radio_GetGroupsEvent"> <xss:annotation> <xss:documentation></xss:documentation> </xss:annotation> <xss:complexType> <xss:complexContent> <xss:extension base="ct:typeEvent"></pre>

```

<xs:sequence>
  <xs:element name="radio" type="ct:typeSubscriberAddress"/>
  <xs:element name="group" type="ctR:typeRadioGroupSelection" minOccurs="0"
maxOccurs="unbounded"/>
  <xs:element name="listEnd" type="xs:boolean"/>
</xs:sequence>
</xs:extension>
</xs:complexContent>
</xs:complexType>
</xs:element>

```

Element Radio_GetGroupsEvent / radio

Namespace	DR-GW-Interface/DR-GW-Radio.Events
Diagram	
Type	ct:typeSubscriberAddress
Properties	content: complex
Model	ct:ssi ct:tsi
Children	ct:ssi, ct:tsi
Instance	<radio xmlns="DR-GW-Interface/DR-GW-Radio.Events" xmlns:ct="DR-GW-Interface/CommonTypes"> <ct:ssi>{1,1}</ct:ssi> <ct:tsi>{1,1}</ct:tsi> </radio>
Source	<xs:element name="radio" type="ct:typeSubscriberAddress"/>

Element Radio_GetGroupsEvent / group

Namespace	DR-GW-Interface/DR-GW-Radio.Events
Diagram	
Type	typeRadioGroupSelection
Properties	content: complex minOccurs: 0 maxOccurs: unbounded
Model	group , level
Children	group, level
Instance	<group xmlns="DR-GW-Interface/DR-GW-Radio.Events" xmlns:ctR="DR-GW-Interface/DR-GW-Radio.CommonTypes"> <ctR:group>{1,1}</ctR:group> <ctR:level>{1,1}</ctR:level> </group>
Source	<xs:element name="group" type="ctR:typeRadioGroupSelection" minOccurs="0" maxOccurs="unbounded"/>

Element Radio_GetGroupsEvent / listEnd

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

Type	xs:boolean
Properties	content: simple
Source	<xs:element name="listEnd" type="xs:boolean"/>

Element Radio_GroupsEvent

Namespace	DR-GW-Interface/DR-GW-Radio.Events
Annotations	
Diagram	<pre> classDiagram ct:typeEvent < -- Radio_GroupsEvent : Extension of 'ct:typeEvent' Radio_GroupsEvent < -- ct:requestId Radio_GroupsEvent < -- ct:result Radio_GroupsEvent -- "1..∞" group : ctR:typeRadioGroupSelection Radio_GroupsEvent -- "1..1" radio : ct:typeSubscriberAddress Radio_GroupsEvent -- "1..1" deletedGroup : ct:typeSubscriberAddress </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} , radio , (group+ deletedGroup)
Children	ct:requestId, ct:result, deletedGroup, group, radio
Instance	<pre> <Radio_GroupsEvent xmlns="DR-GW-Interface/DR-GW-Radio.Events" xmlns:ct="DR-GW-Interface/CommonTypes"> <ct:requestId>{0,1}</ct:requestId> <ct:result>{0,1}</ct:result> <radio>{1,1}</radio> <group>{1,unbounded}</group> <deletedGroup>{1,1}</deletedGroup> </Radio_GroupsEvent> </pre>
Source	<pre> <xs:element name="Radio_GroupsEvent"> <xs:annotation> <xs:documentation/> </xs:annotation> <xs:complexType> <xs:complexContent> <xs:extension base="ct:typeEvent"> <xs:sequence> <xs:element name="radio" type="ct:typeSubscriberAddress"/> <xs:choice> <xs:element name="group" type="ctR:typeRadioGroupSelection" maxOccurs="unbounded"/> <xs:element name="deletedGroup" type="ct:typeSubscriberAddress"/> </xs:choice> </xs:sequence> </xs:extension> </xs:complexContent> </xs:complexType> </xs:element> </pre>

Element Radio_GroupsEvent / radio

Namespace	DR-GW-Interface/DR-GW-Radio.Events
Diagram	<pre> radio : ct:typeSubscriberAddress radio --> ssi radio --> tsi </pre>

Type	ct:typeSubscriberAddress
Properties	content: complex
Model	ct:ssi ct:tsi
Children	ct:ssi, ct:tsi
Instance	<radio xmlns="DR-GW-Interface/DR-GW-Radio.Events" xmlns:ct="DR-GW-Interface/CommonTypes"> <ct:ssi>{1,1}</ct:ssi> <ct:tsi>{1,1}</ct:tsi> </radio>
Source	<xss:element name="radio" type="ct:typeSubscriberAddress"/>

Element Radio_GroupsEvent / group

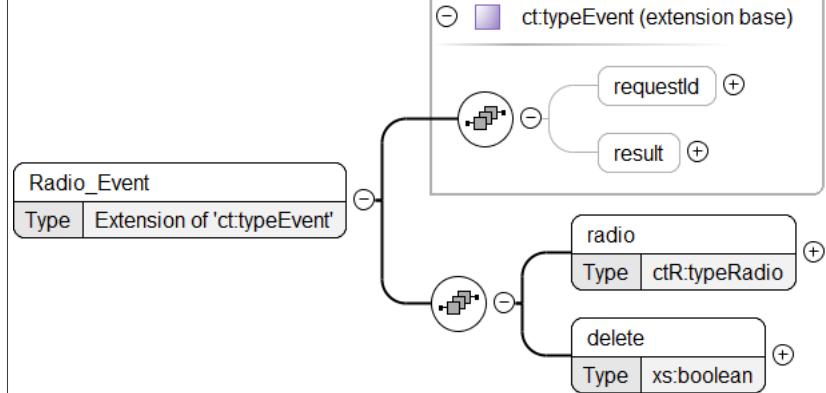
Namespace	DR-GW-Interface/DR-GW-Radio.Events				
Diagram	<pre> classDiagram class group { <<Type ctR:typeRadioGroupSelection>> } class ctR:typeRadioGroupSelection { <<ctR:group>> <<ctR:level>> } group "1..1" -- "1..1" ctR:typeRadioGroupSelection ctR:typeRadioGroupSelection "1..1" -- "1..1" group ctR:typeRadioGroupSelection "1..1" -- "1..1" level level "1..1" -- "1..1" group </pre>				
Type	typeRadioGroupSelection				
Properties	<table border="1"> <tr> <td>content:</td> <td>complex</td> </tr> <tr> <td>maxOccurs:</td> <td>unbounded</td> </tr> </table>	content:	complex	maxOccurs:	unbounded
content:	complex				
maxOccurs:	unbounded				
Model	group , level				
Children	group, level				
Instance	<group xmlns="DR-GW-Interface/DR-GW-Radio.Events" xmlns:ctR="DR-GW-Interface/DR-GW-Radio.CommonTypes"> <ctR:group>{1,1}</ctR:group> <ctR:level>{1,1}</ctR:level> </group>				
Source	<xss:element name="group" type="ctR:typeRadioGroupSelection" maxOccurs="unbounded"/>				

Element Radio_GroupsEvent / deletedGroup

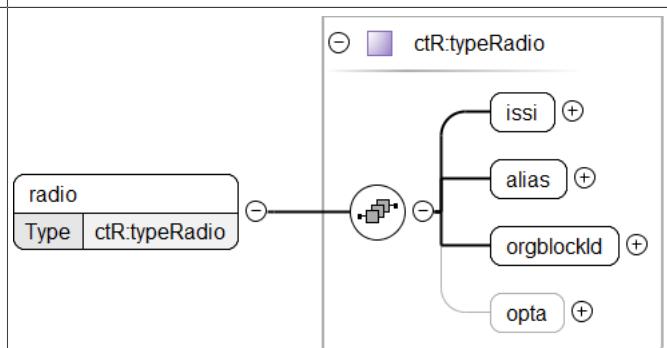
Namespace	DR-GW-Interface/DR-GW-Radio.Events
Diagram	<pre> classDiagram class deletedGroup { <<Type ct:typeSubscriberAddress>> } class ct:typeSubscriberAddress { <<ct:ssi>> <<ct:tsi>> } deletedGroup "1..1" -- "1..1" ct:typeSubscriberAddress ct:typeSubscriberAddress "1..1" -- "1..1" deletedGroup ct:typeSubscriberAddress "1..1" -- "1..1" ssi ct:typeSubscriberAddress "1..1" -- "1..1" tsi </pre>
Type	ct:typeSubscriberAddress
Properties	content: complex
Model	ct:ssi ct:tsi
Children	ct:ssi, ct:tsi
Instance	<deletedGroup xmlns="DR-GW-Interface/DR-GW-Radio.Events" xmlns:ct="DR-GW-Interface/CommonTypes"> <ct:ssi>{1,1}</ct:ssi> <ct:tsi>{1,1}</ct:tsi> </deletedGroup>
Source	<xss:element name="deletedGroup" type="ct:typeSubscriberAddress"/>

Element Radio_Event

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

Diagram	
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} , radio , delete
Children	ct:requestId, ct:result, delete, radio
Instance	<pre><Radio_Event xmlns="DR-GW-Interface/DR-GW-Radio.Events" xmlns:ct="DR-GW-Interface/CommonTypes"> <ct:requestId>{0,1}</ct:requestId> <ct:result>{0,1}</ct:result> <radio>{1,1}</radio> <delete>{1,1}</delete> </Radio_Event></pre>
Source	<pre><xss:element name="Radio_Event"> <xss:annotation> <xss:documentation/> </xss:annotation> <xss:complexType> <xss:complexContent> <xss:extension base="ct:typeEvent"> <xss:sequence> <xss:element name="radio" type="ctR:typeRadio"/> <xss:element name="delete" type="xs:boolean"/> </xss:sequence> </xss:extension> </xss:complexContent> </xss:complexType> </xss:element></pre>

Element Radio_Event / radio

Namespace	DR-GW-Interface/DR-GW-Radio.Events
Diagram	
Type	typeRadio
Properties	content: complex
Model	issi , alias , orgblockId , opta{0,1}
Children	alias, issi, opta, orgblockId
Instance	<pre><radio xmlns="DR-GW-Interface/DR-GW-Radio.Events" xmlns:ctR="DR-GW-Interface/DR-GW-Radio.CommonTypes"></pre>

	<pre><ctR:issi>{1,1}</ctR:issi> <ctR:alias>{1,1}</ctR:alias> <ctR:orgblockId>{1,1}</ctR:orgblockId> <ctR:opta>{0,1}</ctR:opta> </radio></pre>
Source	<code><xss:element name="radio" type="ctR:typeRadio"/></code>

Element Radio_Event / delete

Namespace	DR-GW-Interface/DR-GW-Radio.Events
Diagram	
Type	xs:boolean
Properties	content: simple
Source	<code><xss:element name="delete" type="xs:boolean"/></code>

Element Radio_TrackEvent

Namespace	DR-GW-Interface/DR-GW-Radio.Events
Annotations	
Diagram	
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} , trackingData
Children	ct:requestId, ct:result, trackingData
Instance	<pre><Radio_TrackEvent xmlns="DR-GW-Interface/DR-GW-Radio.Events" xmlns:ct="DR-GW-Interface/CommonTypes"> <ct:requestId>{0,1}</ct:requestId> <ct:result>{0,1}</ct:result> <trackingData>{1,1}</trackingData> </Radio_TrackEvent></pre>
Source	<pre><xss:element name="Radio_TrackEvent"> <xss:annotation> <xss:documentation></xss:documentation> </xss:annotation> <xss:complexType> <xss:complexContent> <xss:extension base="ct:typeEvent"> <xss:sequence> <xss:element name="trackingData" type="ctR:typeRadioTrackingData"/> </xss:sequence> </xss:extension> </xss:complexContent> </xss:complexType> </xss:element></pre>

Element Radio_TrackEvent / trackingData

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

Diagram	<pre> classDiagram class trackingData { <<Type: ctR:typeRadioTrackingData>> } class ctR:typeRadioTrackingData { radio registered exchangeId locationArea lastActive scanningOn status callType callParty dmoState emergency } trackingData "0..1" --> "1..1" ctR:typeRadioTrackingData </pre>
Type	typeRadioTrackingData
Properties	content: complex
Model	radio , registered , exchangeId , locationArea{0,1} , lastActive , scanningOn , status , callType , callParty , dmoState , emergency
Children	callParty, callType, dmoState, emergency, exchangeId, lastActive, locationArea, radio, registered, scanningOn, status
Instance	<pre> <trackingData xmlns="DR-GW-Interface/DR-GW-Radio.Events" xmlns:ctR="DR-GW-Interface/DR-GW-Radio.CommonTypes"> <ctR:radio>{1,1}</ctR:radio> <ctR:registered>{1,1}</ctR:registered> <ctR:exchangeId>{1,1}</ctR:exchangeId> <ctR:locationArea>{0,1}</ctR:locationArea> <ctR:lastActive>{1,1}</ctR:lastActive> <ctR:scanningOn>{1,1}</ctR:scanningOn> <ctR:status>{1,1}</ctR:status> <ctR:callType>{1,1}</ctR:callType> <ctR:callParty>{1,1}</ctR:callParty> <ctR:dmoState>{1,1}</ctR:dmoState> <ctR:emergency>{1,1}</ctR:emergency> </trackingData> </pre>
Source	<code><xss:element name="trackingData" type="ctR:typeRadioTrackingData" /></code>

Element Radio_ChangeOPTAEEvent

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

Diagram	<pre> classDiagram ct:typeEvent < -- Radio_ChangeOPTAEVENT Radio_ChangeOPTAEVENT : Extension of 'ct:typeEvent' Radio_ChangeOPTAEVENT "0..1" --> requestId : ct:typeEvent Radio_ChangeOPTAEVENT "0..1" --> result : ct:typeEvent requestId --> radio : ct:typeSubscriberAddress requestId --> opta : ct:typeOPTA </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} , radio , opta
Children	ct:requestId, ct:result, opta, radio
Instance	<pre> <Radio_ChangeOPTAEVENT xmlns="DR-GW-Interface/DR-GW-Radio.Events" xmlns:ct="DR-GW-Interface/CommonTypes"> <ct:requestId>{0,1}</ct:requestId> <ct:result>{0,1}</ct:result> <radio>{1,1}</radio> <opta>{1,1}</opta> </Radio_ChangeOPTAEVENT> </pre>
Source	<pre> <xss:element name="Radio_ChangeOPTAEVENT"> <xss:annotation> <xss:documentation/> </xss:annotation> <xss:complexType> <xss:complexContent> <xss:extension base="ct:typeEvent"> <xss:sequence> <xss:element name="radio" type="ct:typeSubscriberAddress"/> <xss:element name="opta" type="ct:typeOPTA"/> </xss:sequence> </xss:extension> </xss:complexContent> </xss:complexType> </xss:element> </pre>

Element Radio_ChangeOPTAEVENT / radio

Namespace	DR-GW-Interface/DR-GW-Radio.Events
Diagram	<pre> classDiagram ct:typeSubscriberAddress < -- radio radio : ct:typeSubscriberAddress radio "0..1" --> ssi : ct:typeEvent radio "0..1" --> tsi : ct:typeEvent </pre>
Type	ct:typeSubscriberAddress
Properties	content: complex
Model	ct:ssi ct:tsi
Children	ct:ssi, ct:tsi
Instance	<pre> <radio xmlns="DR-GW-Interface/DR-GW-Radio.Events" xmlns:ct="DR-GW-Interface/CommonTypes"> <ct:ssi>{1,1}</ct:ssi> <ct:tsi>{1,1}</ct:tsi> </radio> </pre>
Source	<pre> <xss:element name="radio" type="ct:typeSubscriberAddress"/> </pre>

Element Radio_ChangeOPTAEVENT / opta

Namespace	DR-GW-Interface/DR-GW-Radio.Events
Diagram	<pre> classDiagram opta < -- ct:typeOPTA opta < -- ct:typeEvent opta --> constraint: OPTA string. Maximum length is 24 characters </pre>
Type	ct:typeOPTA
Properties	content: simple
Facets	maxLength 24
Source	<xs:element name="opta" type="ct:typeOPTA" />

Element Radio_EnDisableEvent

Namespace	DR-GW-Interface/DR-GW-Radio.Events
Annotations	
Diagram	<pre> classDiagram Radio_EnDisableEvent < -- ct:typeEvent Radio_EnDisableEvent < -- ct:typeEvent </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} , radio , reason{0,1} , enabled , overTheAir{0,1}
Children	ct:requestId, ct:result, enabled, overTheAir, radio, reason
Instance	<pre> <Radio_EnDisableEvent xmlns="DR-GW-Interface/DR-GW-Radio.Events" xmlns:ct="DR-GW-Interface/CommonTypes"> <t:requestId>{0,1}</t:requestId> <t:result>{0,1}</t:result> <radio>{1,1}</radio> <reason>{0,1}</reason> <enabled>{1,1}</enabled> <overTheAir>{0,1}</overTheAir> </Radio_EnDisableEvent> </pre>
Source	<pre> <xs:element name="Radio_EnDisableEvent"> <xs:annotation> <xs:documentation></xs:documentation> </xs:annotation> <xs:complexType> <xs:complexContent> <xs:extension base="ct:typeEvent"> <xs:sequence> <xs:element name="radio" type="ct:typeSubscriberAddress"/> </xs:sequence> </xs:extension> </xs:complexContent> </xs:complexType> </xs:element> </pre>

```

<xs:element name="radio" type="ct:typeSubscriberAddress" minOccurs="0" />
<xs:element name="enabled" type="xs:boolean" />
<xs:element name="overTheAir" type="xs:boolean" minOccurs="0" />
</xs:sequence>
</xs:extension>
</xs:complexContent>
</xs:complexType>
</xs:element>

```

Element Radio_EnDisableEvent / radio

Namespace	DR-GW-Interface/DR-GW-Radio.Events
Diagram	
Type	ct:typeSubscriberAddress
Properties	content: complex
Model	ct:ssi ct:tsi
Children	ct:ssi, ct:tsi
Instance	<radio xmlns="DR-GW-Interface/DR-GW-Radio.Events" xmlns:ct="DR-GW-Interface/CommonTypes"> <ct:ssi>{1,1}</ct:ssi> <ct:tsi>{1,1}</ct:tsi> </radio>
Source	<xs:element name="radio" type="ct:typeSubscriberAddress" />

Element Radio_EnDisableEvent / reason

Namespace	DR-GW-Interface/DR-GW-Radio.Events
Diagram	
Type	xs:unsignedByte
Properties	content: simple minOccurs: 0
Source	<xs:element name="reason" type="xs:unsignedByte" minOccurs="0" />

Element Radio_EnDisableEvent / enabled

Namespace	DR-GW-Interface/DR-GW-Radio.Events
Diagram	
Type	xs:boolean
Properties	content: simple
Source	<xs:element name="enabled" type="xs:boolean" />

Element Radio_EnDisableEvent / overTheAir

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

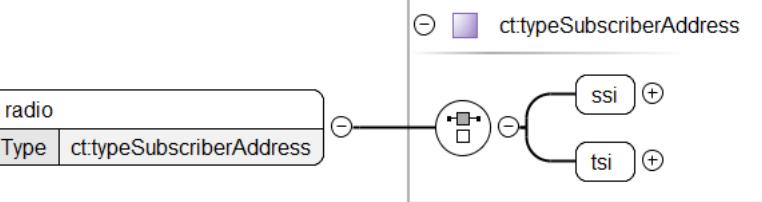
Diagram	<p>The diagram shows the <code>overTheAir</code> element with a type constraint of <code>xs:boolean</code>. A callout box indicates that <code>xs:boolean</code> is a built-in primitive type defining boolean values true and false.</p>				
Type	<code>xs:boolean</code>				
Properties	<table border="1"> <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><xs:element name="overTheAir" type="xs:boolean" minOccurs="0" /></code>				

Element Radio_TrackSubscriptionEvent

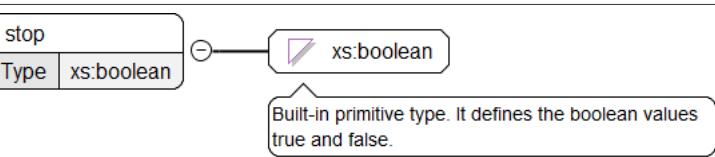
Namespace	DR-GW-Interface/DR-GW-Radio.Events
Annotations	
Diagram	<p>The diagram illustrates the structure of <code>Radio_TrackSubscriptionEvent</code>. It is an extension of <code>ct:typeEvent</code>. It contains attributes <code>requestId</code>, <code>result</code>, <code>radio</code> (of type <code>ct:typeSubscriberAddress</code>), and <code>stop</code> (of type <code>xs:boolean</code>).</p>
Type	extension of <code>ct:typeEvent</code>
Type hierarchy	<ul style="list-style-type: none"> • <code>ct:typeEvent</code>
Properties	content: complex
Model	<code>ct:requestId{0,1}</code> , <code>ct:result{0,1}</code> , <code>radio</code> , <code>stop</code>
Children	<code>ct:requestId</code> , <code>ct:result</code> , <code>radio</code> , <code>stop</code>
Instance	<pre><Radio_TrackSubscriptionEvent xmlns="DR-GW-Interface/DR-GW-Radio.Events" xmlns:ct="DR-GW-Interface/CommonTypes"> <ct:requestId>{0,1}</ct:requestId> <ct:result>{0,1}</ct:result> <radio>{1,1}</radio> <stop>{1,1}</stop> </Radio_TrackSubscriptionEvent></pre>
Source	<pre><xs:element name="Radio_TrackSubscriptionEvent"> <xs:annotation> <xs:documentation></xs:documentation> </xs:annotation> <xs:complexType> <xs:complexContent> <xs:extension base="ct:typeEvent"> <xs:sequence> <xs:element name="radio" type="ct:typeSubscriberAddress"/> <xs:element name="stop" type="xs:boolean"/> </xs:sequence> </xs:extension> </xs:complexContent> </xs:complexType> </xs:element></pre>

Element Radio_TrackSubscriptionEvent / radio

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

Diagram	
Type	ct:typeSubscriberAddress
Properties	content: complex
Model	ct:ssi ct:tsi
Children	ct:ssi, ct:tsi
Instance	<pre><radio xmlns="DR-GW-Interface/DR-GW-Radio.Events" xmlns:ct="DR-GW-Interface/CommonTypes"> <ct:ssi>{1,1}</ct:ssi> <ct:tsi>{1,1}</ct:tsi> </radio></pre>
Source	<pre><xss:element name="radio" type="ct:typeSubscriberAddress"/></pre>

Element Radio_TrackSubscriptionEvent / stop

Namespace	DR-GW-Interface/DR-GW-Radio.Events
Diagram	
Type	xs:boolean
Properties	content: simple
Source	<pre><xss:element name="stop" type="xs:boolean"/></pre>

Namespace: "DR-GW-Interface/CommonTypes"

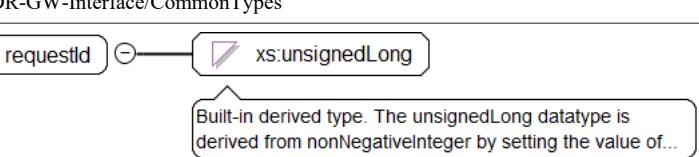
Schema(s)

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><xss:element name="requestId" type="xs:unsignedLong"/></pre>

Element ct:typeResponse / ct:result

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

Diagram	
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													
Type	ct:typeResponseCode												
Properties	content: simple												
Facets	<table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 15%;">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><xss:element name="responseCode" type="ct:typeResponseCode"/></code>												

Element ct:typeResult / ct:sourceSystem

Namespace	DR-GW-Interface/CommonTypes						
Diagram							
Type	ct:typeSourceSystem						
Properties	<table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 15%;">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 style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 15%;">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	
Type	xs:unsignedLong
Properties	content: simple minOccurs: 0
Source	<xs:element name="result" type="xs:unsignedLong" minOccurs="0" />

Element ct:typeEvent / ct:requestId

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

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	<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>
Source	<xs:element name="result" type="ct:typeResult" minOccurs="0" />

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	<pre> classDiagram class tsi { <<Basic type for TETRA subscriber identity containing Network code(MNC) and Country code(MCC).>> <<ct:mnc , ct:mcc , ct:ssi>> mnc mcc ssi } </pre>
Type	ct:typeTSI
Properties	content: complex
Model	ct:mnc , ct:mcc , ct:ssi
Children	ct:mec, ct:mnc, ct:ssi
Instance	<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>
Source	<xs:element name="tsi" type="ct:typeTSI" />

Element ct:typeTSI / ct:mnc

Namespace	DR-GW-Interface/CommonTypes
Diagram	<pre> classDiagram class mnc { <<Built-in derived type. The unsignedShort datatype is derived from unsignedInt by setting the value of maxInclusive to...>> <<xs:unsignedShort>> } </pre>
Type	xs:unsignedShort
Properties	content: simple
Source	<xs:element name="mnc" type="xs:unsignedShort" />

Element ct:typeTSI / ct:mcc

Namespace	DR-GW-Interface/CommonTypes
Diagram	<pre> classDiagram class mcc { <<Built-in derived type. The unsignedShort datatype is derived from unsignedInt by setting the value of maxInclusive to...>> <<xs:unsignedShort>> } </pre>
Type	xs:unsignedShort
Properties	content: simple
Source	<xs:element name="mcc" type="xs:unsignedShort" />

Element ct:typeTSI / ct:ssi

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

Diagram	
Type	xs:unsignedLong
Properties	content: simple
Source	<code><xs:element name="ssi" type="xs:unsignedLong" /></code>

Element ct:typeExternal / ct:gatewayNumber

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

Element ct:typeExternal / ct:number

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

Element ct:typeAddress / ct:subscriber

Namespace	DR-GW-Interface/CommonTypes
Diagram	
Type	ct:typeSubscriberAddress
Properties	content: complex minOccurs: 0
Model	ct:ssi ct:tsi
Children	ct:ssi, ct:tsi
Instance	<code><ct:subscriber xmlns:ct="DR-GW-Interface/CommonTypes"> <ct:ssi>{1,1}</ct:ssi> <ct:tsi>{1,1}</ct:tsi> </ct:subscriber></code>
Source	<code><xs:element name="subscriber" type="ct:typeSubscriberAddress" minOccurs="0" /></code>

Element ct:typeAddress / ct:alias

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

Diagram	
	Built-in derived type. The normalizedString datatype represents white space normalized strings. The base type of...
Type	xs:normalizedString
Properties	content: simple minOccurs: 0
Source	<code><xs:element name="alias" type="xs:normalizedString" minOccurs="0"/></code>

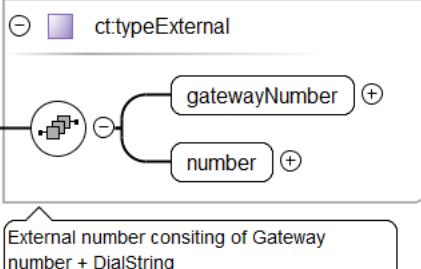
Element ct:typeAddress / ct:msisdn

Namespace	DR-GW-Interface/CommonTypes
Diagram	
	Allowed characters are digits 0 - 9, *, #, A, B, C and D. Maximum length is 24 characters.
Type	ct:typeDialString
Properties	content: simple minOccurs: 0
Facets	maxLength 24
Source	<code><xs: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	
	Fleet specific short number Built-in derived type. The unsignedLong datatype is derived from nonNegativeInteger by setting the value of...
Type	xs:unsignedLong
Properties	content: simple minOccurs: 0
Source	<code><xs:element name="fssn" type="xs:unsignedLong" minOccurs="0"> <xs:annotation> <xs:documentation>Fleet specific short number</xs:documentation> </xs:annotation> </xs:element></code>

Element ct:typeAddress / ct:external

Namespace	DR-GW-Interface/CommonTypes
Diagram	
	External number consisting of Gateway number + DialString
Type	ct:typeExternal

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

Element ct:typeAddress / ct:opta

Namespace	DR-GW-Interface/CommonTypes
Diagram	<pre> graph LR opta((opta)) --> cttypeOPTA[ct:typeOPTA] subgraph Info [] direction TB cttypeOPTA OPTAString["OPTA string. Maximum length is 24 characters."] OPTAString --- cttypeOPTA end </pre>
Type	ct:typeOPTA
Properties	content: simple minOccurs: 0
Facets	maxLength 24
Source	<xss:element name="opta" type="ct:typeOPTA" minOccurs="0" />

Element ct:typeAddress / ct:cell

Namespace	DR-GW-Interface/CommonTypes
Diagram	<pre> graph LR cell((cell)) --> xsshort[xs:short] subgraph Info [] direction TB xsshort BuiltIn["Built-in derived type. The short datatype is derived from int by setting the value of maxInclusive to be 32767 and..."] BuiltIn --- xsshort end </pre>
Type	xs:short
Properties	content: simple minOccurs: 0
Source	<xss:element name="cell" type="xs:short" minOccurs="0" />

Element ct:typeRequest / ct:requestId

Namespace	DR-GW-Interface/CommonTypes
Diagram	<pre> graph LR requestId((requestId)) --> xsunsignedLong[xs:unsignedLong] subgraph Info [] direction TB xsunsignedLong BuiltIn["Built-in derived type. The unsignedLong datatype is derived from nonNegativeInteger by setting the value of..."] BuiltIn --- xsunsignedLong end </pre>
Type	xs:unsignedLong
Properties	content: simple
Source	<xss: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	
Used by	Element Radio_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	
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" /> <xs:element name="sourceSystem" type="ct:typeSourceSystem" minOccurs="0" /> <xs:element name="result" type="xs:unsignedLong" minOccurs="0" /> </xs:sequence> </xs:complexType></pre>

Complex Type ct:typeEvent

Namespace	DR-GW-Interface/CommonTypes
Diagram	
Used by	Elements Radio_ChangeOPTAEvent, Radio_DisableEvent, Radio_Event, Radio_GetEvent, Radio_GroupsEvent, Radio_GetListEvent, Radio_GroupsEvent, Radio_TrackEvent, Radio_TrackSubscriptionEvent
Model	ct:requestId{0,1} , ct:result{0,1}
Children	ct:requestId, ct:result
Source	<pre><xs:complexType name="typeEvent"> <xs:sequence></pre>

```

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

```

Complex Type ct:typeSubscriberAddress

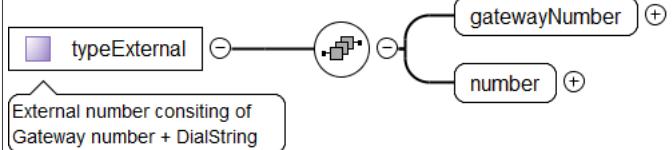
Namespace	DR-GW-Interface/CommonTypes
Annotations	
Diagram	
Used by	Elements Radio_ChangeOPTAEvent/radio, Radio_DisableEvent/radio, Radio_GetGroupsEvent/radio, Radio_GroupsEvent/deletedGroup, Radio_GroupsEvent/radio, Radio_TrackSubscriptionEvent/radio, ct:typeAddress/ct:subscriber, typeRadio/issi, typeRadioGroupSelection/group, typeRadioTrackingData/callParty, typeRadioTrackingData/radio
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:typeTSI

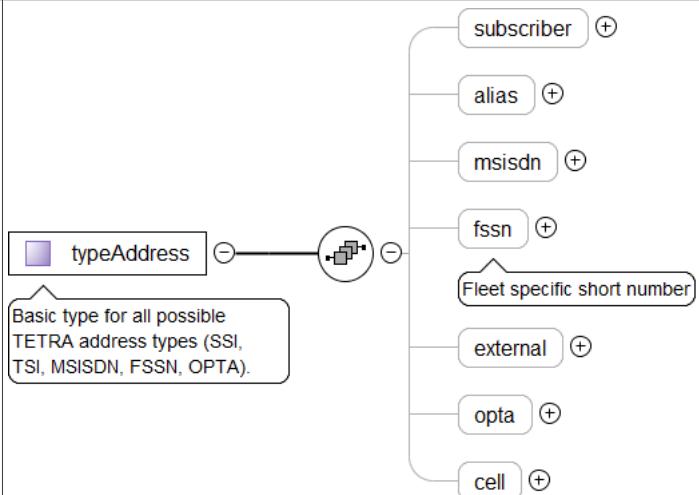
Namespace	DR-GW-Interface/CommonTypes
Annotations	Basic type for TETRA subscriber identity containing Network code(MNC) and Country code(MCC).
Diagram	
Used by	Element ct:typeSubscriberAddress/ct:tsi
Model	ct:mnc , ct:mcc , ct:ssi
Children	ct:mec, ct:mnc, ct:ssi
Source	<pre> <xs:complexType name="typeTSI">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

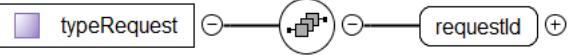
Diagram	
Used by	Element ct:typeAddress/ct:external
Model	ct:gatewayNumber , ct:number
Children	ct:gatewayNumber, ct:number
Source	<pre><xs:complexType name="typeExternal"> <xs:annotation> <xs:documentation>External number consisting of Gateway number + DialString</xs:documentation> </xs:annotation>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

Complex Type ct:typeAddress

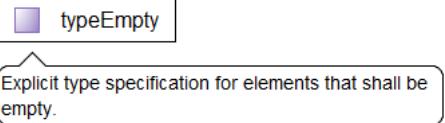
Namespace	DR-GW-Interface/CommonTypes
Annotations	Basic type for all possible TETRA address types (SSI, TSI, MSISDN, FSSN, OPTA).
Diagram	
Model	ct:subscriber{0,1} , ct:alias{0,1} , ct:msisdn{0,1} , ct:fssn{0,1} , ct:external{0,1} , ct:opta{0,1} , ct:cell{0,1}
Children	ct:alias, ct:cell, ct:external,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

Complex Type ct:typeRequest

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

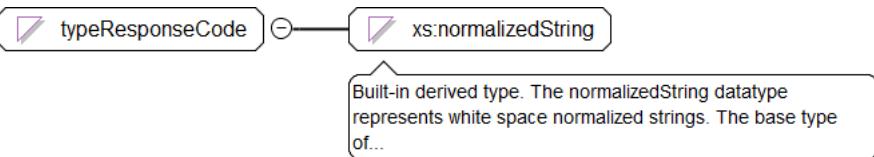
Diagram	
Model	ct:requestId
Children	ct:requestId
Source	<pre><xs:complexType name="typeRequest"> <xs:sequence> <xs:element name="requestId" type="xs:unsignedLong" /> </xs:sequence> </xs:complexType></pre>

Complex Type ct:typeEmpty

Namespace	DR-GW-Interface/CommonTypes
Annotations	Explicit type specification for elements that shall be empty.
Diagram	
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													
Type	restriction of xs:normalizedString												
Facets	<table border="0"> <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: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></pre>												

Simple Type ct:typeSourceSystem

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

Diagram	
Type	restriction of xs:normalizedString
Facets	enumeration DR-GW
	enumeration TCS-API
	enumeration TETRA
Used by	Element ct:typeResult/ct:sourceSystem
Source	<pre><xs:simpleType name="typeSourceSystem"> <xs:restriction base="xs:normalizedString"> <xs:enumeration value="DR-GW"/> <xs:enumeration value="TCS-API"/> <xs:enumeration value="TETRA"/> </xs:restriction> </xs:simpleType></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	Elements Radio_ChangeOPTAEVENT/opta, ct:typeAddress/ct:opta, typeLastKnownOPTA/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: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><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"></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 < -- xs:normalizedString typeAddressingStyle "1" -- "0..1" ucast typeAddressingStyle "1" -- "0..1" mcast </pre> <p>Describes the IP addressing style. Unicast or multicast.</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	<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> <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>				

Namespace: "DR-GW-Interface/DR-GW-Radio.CommonTypes"

Schema(s)

Imported schema DR-GW-Radio.CommonTypes.xsd

Namespace	DR-GW-Interface/DR-GW-Radio.CommonTypes				
Annotations	Version 1.1.1				
Properties	<table> <tr> <td>attribute form default:</td> <td>unqualified</td> </tr> <tr> <td>element form default:</td> <td>qualified</td> </tr> </table>	attribute form default:	unqualified	element form default:	qualified
attribute form default:	unqualified				
element form default:	qualified				

Element(s)

Element typeRadio / issi

Namespace	DR-GW-Interface/DR-GW-Radio.CommonTypes
Diagram	<pre> classDiagram issi "1" -- "0..1" ct:typeSubscriberAddress issi "1" -- "0..1" ssi issi "1" -- "0..1" tsi </pre>
Type	ct:typeSubscriberAddress
Properties	content: complex
Model	ct:ssi ct:tsi
Children	ct:ssi, ct:tsi
Instance	<pre> <issi xmlns="DR-GW-Interface/DR-GW-Radio.CommonTypes" xmlns:ct="DR-GW-Interface/CommonTypes"> <ct:ssi>{1,1}</ct:ssi> <ct:tsi>{1,1}</ct:tsi> </issi> </pre>
Source	<pre> <xs:element name="issi" type="ct:typeSubscriberAddress" /> </pre>

Element typeRadio / alias

Namespace	DR-GW-Interface/DR-GW-Radio.CommonTypes
Diagram	<pre> alias ⊕--> xs:normalizedString +--- Built-in derived type. The normalizedString datatype represents white space normalized strings. The base type of... </pre>
Type	xs:normalizedString
Properties	content: simple
Source	<xs:element name="alias" type="xs:normalizedString"/>

Element typeRadio / orgblockId

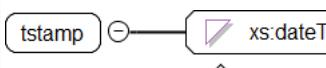
Namespace	DR-GW-Interface/DR-GW-Radio.CommonTypes
Diagram	<pre> typeOrganisationBlockId ⊕--> orgblockId orgblockId ⊕--> orgblockId orgblockId ⊕--> orgblockIdSimple </pre>
Type	typeOrganisationBlockId
Properties	content: complex
Model	orgblockId orgblockIdSimple
Children	orgblockId, orgblockIdSimple
Instance	<orgblockId xmlns="DR-GW-Interface/DR-GW-Radio.CommonTypes" xmlns:ctO="DR-GW-Interface/DR-GW-OrganisationBlock.CommonTypes"> <ctO:orgblockId>{1,1}</ctO:orgblockId> <ctO:orgblockIdSimple>{1,1}</ctO:orgblockIdSimple> </orgblockId>
Source	<xs:element name="orgblockId" type="ctO:typeOrganisationBlockId"/>

Element typeRadio / opta

Namespace	DR-GW-Interface/DR-GW-Radio.CommonTypes				
Diagram	<pre> typeLastKnownOPTA ⊕--> opta opta ⊕--> tstamp opta ⊕--> opta </pre>				
Type	typeLastKnownOPTA				
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	tstamp , opta				
Children	opta, tstamp				
Instance	<opta xmlns="DR-GW-Interface/DR-GW-Radio.CommonTypes"> <tstamp>{1,1}</tstamp> <opta>{1,1}</opta> </opta>				
Source	<xs:element name="opta" type="typeLastKnownOPTA" minOccurs="0"/>				

Element typeLastKnownOPTA / tstamp

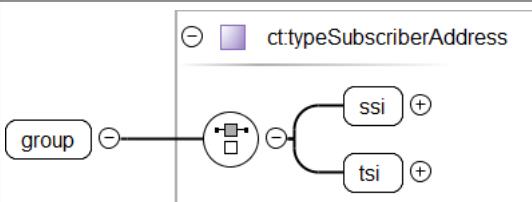
Namespace	DR-GW-Interface/DR-GW-Radio.CommonTypes
-----------	---

Diagram	 Built-in primitive type. The dateTime datatype represents a specific instant of time.
Type	xs:dateTime
Properties	content: simple
Source	<xs:element name="tstamp" type="xs:dateTime" />

Element typeLastKnownOPTA / opta

Namespace	DR-GW-Interface/DR-GW-Radio.CommonTypes
Diagram	 OPTA string. Maximum length is 24 characters.
Type	ct:typeOPTA
Properties	content: simple
Facets	maxLength 24
Source	<xs:element name="opta" type="ct:typeOPTA" />

Element typeRadioGroupSelection / group

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

Element typeRadioGroupSelection / level

Namespace	DR-GW-Interface/DR-GW-Radio.CommonTypes
Diagram	 Covers tcsScanningPriority_t of the TCS-API.
Type	typeGroupSelectionLevel
Properties	content: simple
Facets	enumeration notScanned enumeration low enumeration normal enumeration selected enumeration high

	enumeration	background
Source	<xs:element name="level" type="typeGroupSelectionLevel"/>	

Element typeRadioTrackingData / radio

Namespace	DR-GW-Interface/DR-GW-Radio.CommonTypes
Diagram	<pre> classDiagram class radio class ct:typeSubscriberAddress class ssi class tsi radio "1..1" *-- "++" ct:typeSubscriberAddress ct:typeSubscriberAddress "++" --> "1..1" ssi ct:typeSubscriberAddress "++" --> "1..1" tsi </pre>
Type	ct:typeSubscriberAddress
Properties	content: complex
Model	ct:ssi ct:tsi
Children	ct:ssi, ct:tsi
Instance	<radio xmlns="DR-GW-Interface/DR-GW-Radio.CommonTypes" xmlns:ct="DR-GW-Interface/CommonTypes"> <ct:ssi>{1,1}</ct:ssi> <ct:tsi>{1,1}</ct:tsi> </radio>
Source	<xs:element name="radio" type="ct:typeSubscriberAddress"/>

Element typeRadioTrackingData / registered

Namespace	DR-GW-Interface/DR-GW-Radio.CommonTypes
Diagram	
Type	xs:boolean
Properties	content: simple
Source	<xs:element name="registered" type="xs:boolean"/>

Element typeRadioTrackingData / exchangeId

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

Element typeRadioTrackingData / locationArea

Namespace	DR-GW-Interface/DR-GW-Radio.CommonTypes
Diagram	
Type	xs:unsignedShort

Properties	content: simple minOccurs: 0
Source	<xs:element name="locationArea" type="xs:unsignedShort" minOccurs="0"/>

Element typeRadioTrackingData / lastActive

Namespace	DR-GW-Interface/DR-GW-Radio.CommonTypes
Diagram	<p>Built-in primitive type. The dateTime datatype represents a specific instant of time.</p>
Type	xs:dateTime
Properties	content: simple
Source	<xs:element name="lastActive" type="xs:dateTime"/>

Element typeRadioTrackingData / scanningOn

Namespace	DR-GW-Interface/DR-GW-Radio.CommonTypes
Diagram	<p>Built-in primitive type. It defines the boolean values true and false.</p>
Type	xs:boolean
Properties	content: simple
Source	<xs:element name="scanningOn" type="xs:boolean"/>

Element typeRadioTrackingData / status

Namespace	DR-GW-Interface/DR-GW-Radio.CommonTypes
Diagram	
Type	typeStatusIndicator
Properties	content: complex
Model	value , time
Children	time, value
Instance	<status xmlns="DR-GW-Interface/DR-GW-Radio.CommonTypes"> <value>{1,1}</value> <time>{1,1}</time> </status>
Source	<xs:element name="status" type="typeStatusIndicator"/>

Element typeStatusIndicator / value

Namespace	DR-GW-Interface/DR-GW-Radio.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	<xs:element name="value" type="xs:unsignedLong" />

Element typeStatusIndicator / time

Namespace	DR-GW-Interface/DR-GW-Radio.CommonTypes
Diagram	<pre> graph LR time((time)) --> xsDateTime(xs:dateTime) subgraph Note [] BuiltInPrimitiveType[Built-in primitive type. The dateTime datatype represents a specific instant of time.] end xsDateTime --- Note </pre>
Type	xs:dateTime
Properties	content: simple
Source	<xs:element name="time" type="xs:dateTime" />

Element typeRadioTrackingData / callType

Namespace	DR-GW-Interface/DR-GW-Radio.CommonTypes						
Diagram	<pre> graph LR callType((callType)) --> typeCallType(typeCallType) </pre>						
Type	typeCallType						
Properties	content: simple						
Facets	<table border="1"> <tr> <td>enumeration</td> <td>unknown</td> </tr> <tr> <td>enumeration</td> <td>no</td> </tr> <tr> <td>enumeration</td> <td>individual</td> </tr> </table>	enumeration	unknown	enumeration	no	enumeration	individual
enumeration	unknown						
enumeration	no						
enumeration	individual						
Source	<xs:element name="callType" type="typeCallType" />						

Element typeRadioTrackingData / callParty

Namespace	DR-GW-Interface/DR-GW-Radio.CommonTypes
Diagram	<pre> graph LR callParty((callParty)) --> ctTypeSubscriberAddress(ct:typeSubscriberAddress) ctTypeSubscriberAddress --> ssi(ssi) ctTypeSubscriberAddress --> tsi(tsi) subgraph Note [] ctTypeSubscriberAddress end </pre>
Type	ct:typeSubscriberAddress
Properties	content: complex
Model	ct:ssi ct:tsi
Children	ct:ssi, ct:tsi
Instance	<pre> <callParty xmlns="DR-GW-Interface/DR-GW-Radio.CommonTypes" xmlns:ct="DR-GW-Interface/CommonTypes"> <ct:ssi>{1,1}</ct:ssi> <ct:tsi>{1,1}</ct:tsi> </callParty> </pre>
Source	<xs:element name="callParty" type="ct:typeSubscriberAddress" />

Element typeRadioTrackingData / dmoState

Namespace	DR-GW-Interface/DR-GW-Radio.CommonTypes
Diagram	<pre> graph LR dmoState((dmoState)) --> xsBoolean(xs:boolean) subgraph Note [] BuiltInPrimitiveType[Built-in primitive type. It defines the boolean values true and false.] end xsBoolean --- Note </pre>

Type	xs:boolean
Properties	content: simple
Source	<xs:element name="dmoState" type="xs:boolean" />

Element typeRadioTrackingData / emergency

Namespace	DR-GW-Interface/DR-GW-Radio.CommonTypes
Diagram	
Type	xs:boolean
Properties	content: simple
Source	<xs:element name="emergency" type="xs:boolean" />

Complex Type(s)

Complex Type typeRadio

Namespace	DR-GW-Interface/DR-GW-Radio.CommonTypes
Annotations	
Diagram	
Used by	Elements Radio_Event/radio, Radio_GetEvent/radio, Radio_GetListEvent/radio
Model	issi , alias , orgblockId , opta{0,1}
Children	alias, issi, opta, orgblockId
Source	<pre> <xs:complexType name="typeRadio"> <xs:annotation> <xs:documentation/> </xs:annotation> <xs:sequence> <xs:element name="issi" type="ct:typeSubscriberAddress" /> <xs:element name="alias" type="xs:normalizedString" /> <xs:element name="orgblockId" type="cto:typeOrganisationBlockId" /> <xs:element name="opta" type="typeLastKnownOPTA" minOccurs="0" /> </xs:sequence> </xs:complexType> </pre>

Complex Type typeLastKnownOPTA

Namespace	DR-GW-Interface/DR-GW-Radio.CommonTypes
Diagram	
Used by	Element typeRadio/opta
Model	tstamp , opta
Children	opta, tstamp
Source	<pre> <xs:complexType name="typeLastKnownOPTA"> <xs:sequence> <xs:element name="tstamp" type="xs:dateTime" /> <xs:element name="opta" type="ct:typeOPTA" /> </xs:sequence> </xs:complexType> </pre>

```
</xs:sequence>
</xs:complexType>
```

Complex Type typeRadioGroupSelection

Namespace	DR-GW-Interface/DR-GW-Radio.CommonTypes
Annotations	
Diagram	<pre> classDiagram class typeRadioGroupSelection class group class level typeRadioGroupSelection "1" -- "1" group typeRadioGroupSelection "1" -- "1" level </pre>
Used by	Elements Radio_GetGroupsEvent/group, Radio_GroupsEvent/group
Model	group , level
Children	group, level
Source	<pre> <xs:complexType name="typeRadioGroupSelection"> <xs:annotation> <xs:documentation/> </xs:annotation> <xs:sequence> <xs:element name="group" type="ct:typeSubscriberAddress"/> <xs:element name="level" type="typeGroupSelectionLevel"/> </xs:sequence> </xs:complexType> </pre>

Complex Type typeRadioTrackingData

Namespace	DR-GW-Interface/DR-GW-Radio.CommonTypes
Annotations	
Diagram	<pre> classDiagram class typeRadioTrackingData class radio class registered class exchangeId class locationArea class lastActive class scanningOn class status class callType class callParty class dmoState class emergency typeRadioTrackingData "1" -- "1" radio typeRadioTrackingData "1" -- "1" registered typeRadioTrackingData "1" -- "1" exchangeId typeRadioTrackingData "1" -- "1" locationArea typeRadioTrackingData "1" -- "1" lastActive typeRadioTrackingData "1" -- "1" scanningOn typeRadioTrackingData "1" -- "1" status typeRadioTrackingData "1" -- "1" callType typeRadioTrackingData "1" -- "1" callParty typeRadioTrackingData "1" -- "1" dmoState typeRadioTrackingData "1" -- "1" emergency </pre>
Used by	Element Radio_TrackEvent/trackingData
Model	radio , registered , exchangeId , locationArea{0,1} , lastActive , scanningOn , status , callType , callParty , dmoState , emergency
Children	callParty, callType, dmoState, emergency, exchangeId, lastActive, locationArea, radio, registered, scanningOn, status
Source	<pre> <xs:complexType name="typeRadioTrackingData"> <xs:annotation> <xs:documentation/> </xs:annotation> <xs:sequence> <xs:element name="radio" type="ct:typeSubscriberAddress"/> <xs:element name="registered" type="xs:boolean"/> </xs:sequence> </xs:complexType> </pre>

```

<xs:element name="exchangeId" type="xs:unsignedLong" />
<xs:element name="locationArea" type="xs:unsignedShort" minOccurs="0" />
<xs:element name="lastActive" type="xs:dateTime" />
<xs:element name="scanningOn" type="xs:boolean" />
<xs:element name="status" type="typeStatusIndicator" />
<xs:element name="callType" type="typeCallType" />
<xs:element name="callParty" type="ct:typeSubscriberAddress" />
<xs:element name="dmoState" type="xs:boolean" />
<xs:element name="emergency" type="xs:boolean" />
</xs:sequence>
</xs:complexType>

```

Complex Type `typeStatusIndicator`

Namespace	DR-GW-Interface/DR-GW-Radio.CommonTypes
Annotations	
Diagram	<pre> classDiagram class typeStatusIndicator class value class time typeStatusIndicator "1" -- "1" value : value typeStatusIndicator "1" -- "1" time : time </pre>
Used by	Element typeRadioTrackingData/status
Model	value , time
Children	time, value
Source	<pre> <xs:complexType name="typeStatusIndicator"> <xs:annotation> <xs:documentation/> </xs:annotation> <xs:sequence> <xs:element name="value" type="xs:unsignedLong" /> <xs:element name="time" type="xs:dateTime" /> </xs:sequence> </xs:complexType> </pre>

Simple Type(s)

Simple Type `typeGroupSelectionLevel`

Namespace	DR-GW-Interface/DR-GW-Radio.CommonTypes												
Annotations	Covers <code>tcsScanningPriority_t</code> of the TCS-API.												
Diagram	<pre> classDiagram class typeGroupSelectionLevel class xs_normalizedString typeGroupSelectionLevel "1" -- "1" xs_normalizedString : note over typeGroupSelectionLevel: Covers tcsScanningPriority_t of the TCS-API. note over xs_normalizedString: Built-in derived type. The normalizedString datatype represents white space normalized strings. The base type of... </pre>												
Type	restriction of <code>xs:normalizedString</code>												
Facets	<table border="1"> <tr> <td>enumeration</td> <td>notScanned</td> </tr> <tr> <td>enumeration</td> <td>low</td> </tr> <tr> <td>enumeration</td> <td>normal</td> </tr> <tr> <td>enumeration</td> <td>selected</td> </tr> <tr> <td>enumeration</td> <td>high</td> </tr> <tr> <td>enumeration</td> <td>background</td> </tr> </table>	enumeration	notScanned	enumeration	low	enumeration	normal	enumeration	selected	enumeration	high	enumeration	background
enumeration	notScanned												
enumeration	low												
enumeration	normal												
enumeration	selected												
enumeration	high												
enumeration	background												
Used by	Element typeRadioGroupSelection/level												
Source	<pre> <xs:simpleType name="typeGroupSelectionLevel"> <xs:annotation> <xs:documentation>Covers <code>tcsScanningPriority_t</code> of the TCS-API.</xs:documentation> </xs:annotation> <xs:restriction base="xs:normalizedString"> <xs:enumeration value="notScanned" /> <xs:enumeration value="low" /> <xs:enumeration value="normal" /> <xs:enumeration value="selected" /> <xs:enumeration value="high" /> <xs:enumeration value="background" /> </xs:restriction> </xs:simpleType> </pre>												

</xs:simpleType>

Simple Type typeCallType

Namespace	DR-GW-Interface/DR-GW-Radio.CommonTypes						
Annotations							
Diagram	<pre> graph LR typeCallType --> xs:normalizedString typeCallType -.-> Built-in derived type. The normalizedString datatype represents white space normalized strings. The base type of... note </pre>						
Type	restriction of xs:normalizedString						
Facets	<table border="1"> <tr> <td>enumeration</td> <td>unknown</td> </tr> <tr> <td>enumeration</td> <td>no</td> </tr> <tr> <td>enumeration</td> <td>individual</td> </tr> </table>	enumeration	unknown	enumeration	no	enumeration	individual
enumeration	unknown						
enumeration	no						
enumeration	individual						
Used by	Element typeRadioTrackingData/callType						
Source	<pre> <xs:simpleType name="typeCallType"> <xs:annotation> <xs:documentation/> </xs:annotation> <xs:restriction base="xs:normalizedString"> <xs:enumeration value="unknown"/> <xs:enumeration value="no"/> <xs:enumeration value="individual"/> </xs:restriction> </xs:simpleType> </pre>						

Namespace: "DR-GW-Interface/DR-GW-OrganisationBlock.CommonTypes"

Schema(s)

Imported schema DR-GW-OrganisationBlock.CommonTypes.xsd

Namespace	DR-GW-Interface/DR-GW-OrganisationBlock.CommonTypes				
Annotations	Version 1.1.1				
Properties	<table border="1"> <tr> <td>attribute form default:</td> <td>unqualified</td> </tr> <tr> <td>element form default:</td> <td>qualified</td> </tr> </table>	attribute form default:	unqualified	element form default:	qualified
attribute form default:	unqualified				
element form default:	qualified				

Element(s)

Element typeOrganisationBlockId / orgblockId

Namespace	DR-GW-Interface/DR-GW-OrganisationBlock.CommonTypes
Diagram	<pre> graph LR orgblockId --> typeOrganisationBlockIdNormal typeOrganisationBlockIdNormal -.-> id1, id2, id3, id4, id5, id6 </pre>

Type	typeOrganisationBlockIdNormal
Properties	content: complex
Model	id1{0,1} , id2{0,1} , id3{0,1} , id4{0,1} , id5{0,1} , id6{0,1}
Children	id1, id2, id3, id4, id5, id6
Instance	<pre><orgblockId xmlns="DR-GW-Interface/DR-GW-OrganisationBlock.CommonTypes"> <id1>{0,1}</id1> <id2>{0,1}</id2> <id3>{0,1}</id3> <id4>{0,1}</id4> <id5>{0,1}</id5> <id6>{0,1}</id6> </orgblockId></pre>
Source	<code><x:element name="orgblockId" type="typeOrganisationBlockIdNormal" /></code>

Element typeOrganisationBlockIdNormal / id1

Namespace	DR-GW-Interface/DR-GW-OrganisationBlock.CommonTypes				
Diagram	<pre> classDiagram class id1 { <<xs:unsignedShort>> } id1 --o xs:unsignedShort </pre> <p>Built-in derived type. The unsignedShort datatype is derived from unsignedInt by setting the value of maxInclusive to...</p>				
Type	xs:unsignedShort				
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><x:element name="id1" type="xs:unsignedShort" minOccurs="0" /></code>				

Element typeOrganisationBlockIdNormal / id2

Namespace	DR-GW-Interface/DR-GW-OrganisationBlock.CommonTypes				
Diagram	<pre> classDiagram class id2 { <<xs:unsignedShort>> } id2 --o xs:unsignedShort </pre> <p>Built-in derived type. The unsignedShort datatype is derived from unsignedInt by setting the value of maxInclusive to...</p>				
Type	xs:unsignedShort				
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><x:element name="id2" type="xs:unsignedShort" minOccurs="0" /></code>				

Element typeOrganisationBlockIdNormal / id3

Namespace	DR-GW-Interface/DR-GW-OrganisationBlock.CommonTypes				
Diagram	<pre> classDiagram class id3 { <<xs:unsignedShort>> } id3 --o xs:unsignedShort </pre> <p>Built-in derived type. The unsignedShort datatype is derived from unsignedInt by setting the value of maxInclusive to...</p>				
Type	xs:unsignedShort				
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><x:element name="id3" type="xs:unsignedShort" minOccurs="0" /></code>				

Element typeOrganisationBlockIdNormal / id4

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

Diagram	 Built-in derived type. The unsignedShort datatype is derived from unsignedInt by setting the value of maxInclusive to...
Type	xs:unsignedShort
Properties	content: simple minOccurs: 0
Source	<code><xs:element name="id4" type="xs:unsignedShort" minOccurs="0" /></code>

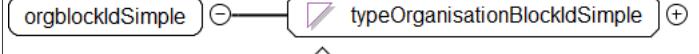
Element typeOrganisationBlockIdNormal / id5

Namespace	DR-GW-Interface/DR-GW-OrganisationBlock.CommonTypes
Diagram	 Built-in derived type. The unsignedShort datatype is derived from unsignedInt by setting the value of maxInclusive to...
Type	xs:unsignedShort
Properties	content: simple minOccurs: 0
Source	<code><xs:element name="id5" type="xs:unsignedShort" minOccurs="0" /></code>

Element typeOrganisationBlockIdNormal / id6

Namespace	DR-GW-Interface/DR-GW-OrganisationBlock.CommonTypes
Diagram	 Built-in derived type. The unsignedShort datatype is derived from unsignedInt by setting the value of maxInclusive to...
Type	xs:unsignedShort
Properties	content: simple minOccurs: 0
Source	<code><xs:element name="id6" type="xs:unsignedShort" minOccurs="0" /></code>

Element typeOrganisationBlockId / orgblockIdSimple

Namespace	DR-GW-Interface/DR-GW-OrganisationBlock.CommonTypes
Diagram	 Organisation block send as simple normalized string. The pattern is: id1-id2-id3-id4-id5-id6
Type	typeOrganisationBlockIdSimple
Properties	content: simple
Facets	pattern <code>(([0-9] [1-9]\d{0,3} [1-5]\d{4} 6[0-4]\d{3} 65[0-4]\d{2} 655[0-2]\d 6553[0-5]) - {0,5} ([0-9] [1-9]\d{0,3} [1-5]\d{4} 6[0-4]\d{3} 65[0-4]\d{2} 655[0-2]\d 6553[0-5]))</code>
Source	<code><xs:element name="orgblockIdSimple" type="typeOrganisationBlockIdSimple" /></code>

Element typeOrganisationBlock / orgblockId

Namespace	DR-GW-Interface/DR-GW-OrganisationBlock.CommonTypes
Diagram	<pre> classDiagram typeOrganisationBlockId { orgblockId orgblockIdSimple } orgblockId < -- typeOrganisationBlockId orgblockIdSimple < -- typeOrganisationBlockId </pre>
Type	typeOrganisationBlockId
Properties	content: complex
Model	orgblockId orgblockIdSimple
Children	orgblockId, orgblockIdSimple
Instance	<pre> <orgblockId xmlns="DR-GW-Interface/DR-GW-OrganisationBlock.CommonTypes"> <orgblockId>{1,1}</orgblockId> <orgblockIdSimple>{1,1}</orgblockIdSimple> </orgblockId> </pre>
Source	<pre><xss:element name="orgblockId" type="typeOrganisationBlockId"/></pre>

Element typeOrganisationBlock / alias

Namespace	DR-GW-Interface/DR-GW-OrganisationBlock.CommonTypes
Diagram	<pre> alias < -- xs:normalizedString </pre> <p>Built-in derived type. The normalizedString datatype represents white space normalized strings. The base type of...</p>
Type	xs:normalizedString
Properties	content: simple
Source	<pre><xss:element name="alias" type="xs:normalizedString"/></pre>

Complex Type(s)

Complex Type typeOrganisationBlockId

Namespace	DR-GW-Interface/DR-GW-OrganisationBlock.CommonTypes
Annotations	
Diagram	<pre> typeOrganisationBlockId { orgblockId orgblockIdSimple } orgblockId < -- typeOrganisationBlockId orgblockIdSimple < -- typeOrganisationBlockId </pre>
Used by	Elements typeOrganisationBlock/orgblockId, typeRadio/orgblockId
Model	orgblockId orgblockIdSimple
Children	orgblockId, orgblockIdSimple
Source	<pre> <xss:complexType name="typeOrganisationBlockId"> <xss:annotation> <xss:documentation/> </xss:annotation> <xss:choice> <xss:element name="orgblockId" type="typeOrganisationBlockIdNormal"/> <xss:element name="orgblockIdSimple" type="typeOrganisationBlockIdSimple"/> </xss:choice> </xss:complexType> </pre>

Complex Type typeOrganisationBlockIdNormal

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

Annotations	
Diagram	<pre> graph LR typeOrganisationBlockIdNormal["typeOrganisationBlockIdNormal"] --> id1["id1"] typeOrganisationBlockIdNormal --> id2["id2"] typeOrganisationBlockIdNormal --> id3["id3"] typeOrganisationBlockIdNormal --> id4["id4"] typeOrganisationBlockIdNormal --> id5["id5"] typeOrganisationBlockIdNormal --> id6["id6"] </pre>
Used by	Element typeOrganisationBlockId/orgblockId
Model	id1{0,1} , id2{0,1} , id3{0,1} , id4{0,1} , id5{0,1} , id6{0,1}
Children	id1, id2, id3, id4, id5, id6
Source	<pre> <xss:complexType name="typeOrganisationBlockIdNormal"> <xss:annotation> <xss:documentation/> </xss:annotation> <xss:sequence> <xss:element name="id1" type="xs:unsignedShort" minOccurs="0"/> <xss:element name="id2" type="xs:unsignedShort" minOccurs="0"/> <xss:element name="id3" type="xs:unsignedShort" minOccurs="0"/> <xss:element name="id4" type="xs:unsignedShort" minOccurs="0"/> <xss:element name="id5" type="xs:unsignedShort" minOccurs="0"/> <xss:element name="id6" type="xs:unsignedShort" minOccurs="0"/> </xss:sequence> </xss:complexType> </pre>

Complex Type typeOrganisationBlock

Namespace	DR-GW-Interface/DR-GW-OrganisationBlock.CommonTypes
Annotations	
Diagram	<pre> graph LR typeOrganisationBlock["typeOrganisationBlock"] --> orgblockId["orgblockId"] typeOrganisationBlock --> alias["alias"] </pre>
Model	orgblockId , alias
Children	alias, orgblockId
Source	<pre> <xss:complexType name="typeOrganisationBlock"> <xss:annotation> <xss:documentation/> </xss:annotation> <xss:sequence> <xss:element name="orgblockId" type="typeOrganisationBlockId"/> <xss:element name="alias" type="xs:normalizedString"/> </xss:sequence> </xss:complexType> </pre>

Simple Type(s)

Simple Type typeOrganisationBlockIdSimple

Namespace	DR-GW-Interface/DR-GW-OrganisationBlock.CommonTypes
Annotations	Organisation block send as simple normalized string. The pattern is: id1-id2-id3-id4-id5-id6
Diagram	<pre> graph LR typeOrganisationBlockIdSimple["typeOrganisationBlockIdSimple"] --> xsNormalizedString["xs:normalizedString"] </pre> <p>Organisation block send as simple normalized string. The pattern is: id1-id2-id3-id4-id5-id6</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	pattern	<pre>(([0-9] [1-9]\d{0,3} [1-5]\d{4} 6[0-4]\d{3} 65[0-4]\d{2} 655[0-2]\d 6553[0-5])-){0,5}([0-9] [1-9]\d{0,3} [1-5]\d{4} 6[0-4]\d{3} 65[0-4]\d{2} 655[0-2]\d 6553[0-5])</pre>
Used by	Element	typeOrganisationBlockId/orgblockIdSimple
Source		<pre><xssimpleType name="typeOrganisationBlockIdSimple"> <xsoannotation> <xsodocumentation>Organisation block send as simple normalized string. The pattern is: id1-id2- id3-id4-id5-id6</xsodocumentation> </xsoannotation> <xsorestriction base="xsnormalizedString"> <xsopattern value="(([0-9] [1-9]\d{0,3} [1-5]\d{4} 6[0-4]\d{3} 65[0-4]\d{2} 655[0-2]\d 6553[0-5])-){0,5}([0-9] [1-9]\d{0,3} [1-5]\d{4} 6[0-4]\d{3} 65[0-4]\d{2} 655[0-2]\d 6553[0-5])"/> </xsorestriction> </xssimpleType></pre>