Schema documentation for DR-GW-Call.xsd

november 5, 2024

Table of Contents

Namespace: "DR-GW-Interface/DR-GW-Call" 2
- Schema(s) 2
  - Main schema `DR-GW-Call.xsd` 2
- Element(s) 2
  - Element `Call_Select` 2
  - Element `Call_Select / sel` 3
  - Element `Call_Request` 3
  - Element `Call_Request / action` 4
  - Element `Call_Request / attributes` 5
  - Element `Call_Request / callingParty` 6
  - Element `Call_Request / calledParty` 6
  - Element `Call_Request / workstationId` 7
  - Element `Call_PTTRequest` 7
  - Element `Call_PTTRequest / action` 8
  - Element `Call_PTTRequest / attributes` 9
  - Element `Call_PTTRequest / talkingParty` 9
  - Element `Call_PTTRequest / workstationId` 10
  - Element `Call_KeyExchange` 10
  - Element `Call_KeyExchange / action` 11

Namespace: "DR-GW-Interface/CommonTypes" 11
- Schema(s) 11
  - Imported schema `CommonTypes.xsd` 11
- Element(s) 12
  - Element `ct:typeRequest / ct:requestId` 12
  - Element `ct:typeAddress / ct:subscriber` 12
  - Element `ct:typeSubscriberAddress / ct:ssi` 12
  - Element `ct:typeSubscriberAddress / ct:tsi` 12
  - Element `ct:typeTSI / ct:mnc` 13
  - Element `ct:typeTSI / ct:mcc` 13
  - Element `ct:typeTSI / ct:ssi` 13
  - Element `ct:typeAddress / ct:alias` 14
  - Element `ct:typeAddress / ct:msisdn` 14
  - Element `ct:typeAddress / ct:fssn` 14
  - Element `ct:typeAddress / ct:external` 14
  - Element `ct:typeExternal / ct:gatewayNumber` 15
  - Element `ct:typeExternal / ct:number` 15
  - Element `ct:typeAddress / ct:opta` 15
  - Element `ct:typeAddress / ct:cell` 16
  - Element `ct:typeResult / ct:responseCode` 16
  - Element `ct:typeResult / ct:sourceSystem` 16
  - Element `ct:typeResult / ct:result` 16
  - Element `ct:typeResponse / ct:requestId` 17
  - Element `ct:typeResponse / ct:result` 17
  - Element `ct:typeEvent / ct:requestId` 17
  - Element `ct:typeEvent / ct:result` 18
- Complex Type(s) 18
  - Complex Type `ct:typeRequest` 18
  - Complex Type `ct:typeAddress` 18
  - Complex Type `ct:typeSubscriberAddress` 19
  - Complex Type `ct:typeTSI` 19
  - Complex Type `ct:typeExternal` 20
  - Complex Type `ct:typeResult` 20
  - Complex Type `ct:typeResponse` 21
  - Complex Type `ct:typeEvent` 21
  - Complex Type `ct:typeEmpty` 21
- Simple Type(s) 22
  - Simple Type `ct:typeDialString` 22
  - Simple Type `ct:typeOPTA` 22
  - Simple Type `ct:typeResponseCode` 22
  - Simple Type `ct:typeSourceSystem` 23
  - Simple Type `ct:typeAddressingStyle` 23

Namespace: "DR-GW-Interface/DR-GW-Call.CommonTypes" 24

Schema(s) 24
Imported schema DR-GW-Call.CommonTypes.xsd 24
Element(s) 24
Element typeSelection / level 24
Element typeSelection / target 24
Element typeCallAttributes / hook 25
Element typeCallAttributes / mode 25
Element typeCallAttributes / commtype 25
Element typeCallAttributes / priority 25
Element typeCallAttributes / encryption 26
Element typeCallAttributes / ambienceListen 26
Element typeCallAttributes / req2speak 26
Element typeCallAttributes / demandPriority 26
Element typeTxGranted / txGrant 27
Element typeTxGranted / talkingParty 27
Element typeTxGranted / encryption 28
Element typeTxGranted / txPriority 28
Element typeTxGranted / txInterrupt 28
Element typeTxGranted / txRepeat 29
Element typeTxGranted / workstationId 29
Element typeDisconnectCause / protocol 29
Element typeDisconnectCause / code 30
Element typeDisconnectCause / text 30
Complex Type(s) 31
Complex Type typeSelection 31
Complex Type typeCallAttributes 31
Complex Type typeTxGranted 32
Complex Type typeDisconnectCause 33
Simple Type(s) 33
Simple Type typeSelectionLevel 33
Simple Type typeActionRequest 34
Simple Type typeCallMode 35
Simple Type typeCallType 35
Simple Type typeTxDemandPriority 36
Simple Type typeWorkstationId 36
Simple Type typeActionPTTRequest 37
Simple Type typeKeyExchangeAction 37
Simple Type typeActionEvent 38
Simple Type typeTxPriority 38
Simple Type typeTxGrant 39
Simple Type typeAudioCodec 39
Simple Type typeKeyExchangeState 40
Simple Type typeKeyExchangeCode 40
Simple Type typeKeyExchangeText 41
Simple Type typeKeyExchangeTextPriority 41
Simple Type typeKeyManagementTextPriority 41
Simple Type typeUnitInEmergencyType 42
Simple Type typeEmergencyInfo 42

Namespace: "DR-GW-Interface/DR-GW-Call"

Schema(s)

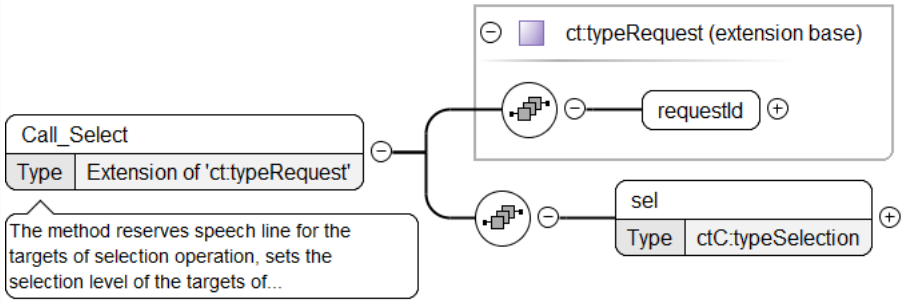
Main schema DR-GW-Call.xsd

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

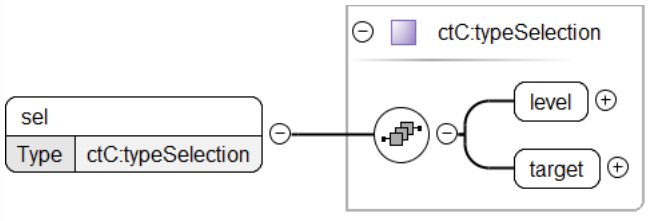
Element(s)

Element Call_Select

Namespace	DR-GW-Interface/DR-GW-Call
Annotations	The method reserves speech line for the targets of selection operation, sets the selection level of the targets of

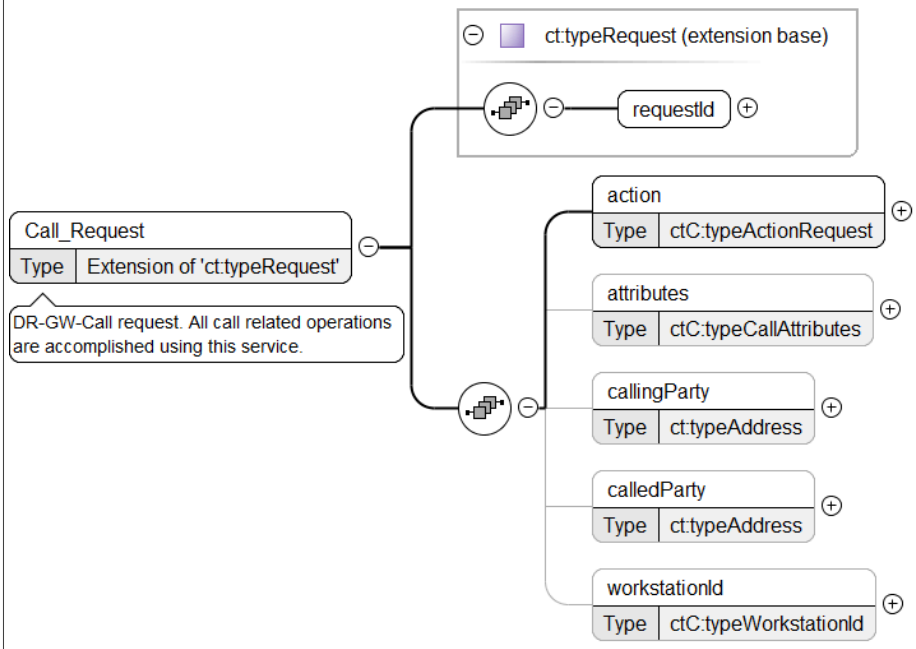
	selection operation to given values.
Diagram	
Type	extension of ct:typeRequest
Type hierarchy	• ct:typeRequest
Properties	content: complex
Model	ct:requestId , sel
Children	ct:requestId, sel
Instance	<Call_Select xmlns="DR-GW-Interface/DR-GW-Call" xmlns:ct="DR-GW-Interface/CommonTypes"> <ct:requestId>{1,1}</ct:requestId> <sel>{1,1}</sel> </Call_Select>
Source	<xs:element name="Call_Select"> <xs:annotation> <xs:documentation>The method reserves speech line for the targets of selection operation, sets the selection level of the targets of selection operation to given values.</xs:documentation> </xs:annotation> <xs:complexType> <xs:complexContent> <xs:extension base="ct:typeRequest"> <xs:sequence> <xs:element name="sel" type="ctC:typeSelection"/> </xs:sequence> </xs:extension> </xs:complexContent> </xs:complexType> </xs:element>

Element Call_Select / sel

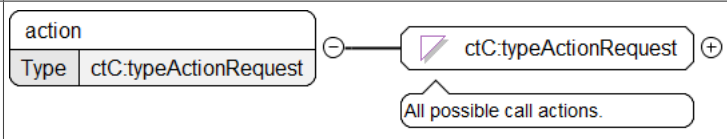
Namespace	DR-GW-Interface/DR-GW-Call
Diagram	
Type	typeSelection
Properties	content: complex
Model	level , target
Children	level, target
Instance	<sel xmlns="DR-GW-Interface/DR-GW-Call" xmlns:ctC="DR-GW-Interface/DR-GW-Call.CommonTypes"> <ctC:level>{1,1}</ctC:level> <ctC:target>{1,1}</ctC:target> </sel>
Source	<xs:element name="sel" type="ctC:typeSelection"/>

Element Call_Request

Namespace	DR-GW-Interface/DR-GW-Call
Annotations	DR-GW-Call request. All call related operations

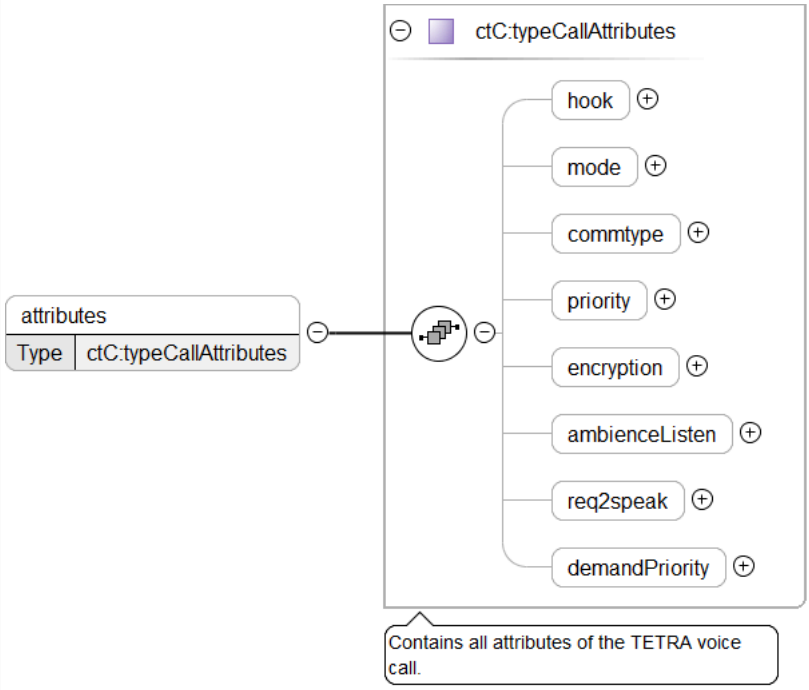
	are accomplished using this service.
Diagram	
Type	extension of ct:typeRequest
Type hierarchy	• ct:typeRequest
Properties	content: complex
Model	ct:requestId , action , attributes{0,1} , callingParty{0,1} , calledParty{0,1} , workstationId{0,1}
Children	action, attributes, calledParty, callingParty, ct:requestId, workstationId
Instance	<Call_Request xmlns="DR-GW-Interface/DR-GW-Call" xmlns:ct="DR-GW-Interface/CommonTypes"> <ct:requestId>{1,1}</ct:requestId> <action>{1,1}</action> <attributes>{0,1}</attributes> <callingParty>{0,1}</callingParty> <calledParty>{0,1}</calledParty> <workstationId>{0,1}</workstationId> </Call_Request>
Source	<xs:element name="Call_Request"> <xs:annotation> <xs:documentation>DR-GW-Call request. All call related operations are accomplished using this service.</xs:documentation> </xs:annotation> <xs:complexType> <xs:complexContent> <xs:extension base="ct:typeRequest"> <xs:sequence> <xs:element name="action" type="ctC:typeActionRequest"/> <xs:element name="attributes" type="ctC:typeCallAttributes" minOccurs="0"/> <xs:element name="callingParty" type="ct:typeAddress" minOccurs="0"/> <xs:element name="calledParty" type="ct:typeAddress" minOccurs="0"/> <xs:element name="workstationId" type="ctC:typeWorkstationId" minOccurs="0"/> </xs:sequence> </xs:extension> </xs:complexContent> </xs:complexType> </xs:element>

Element Call_Request / action

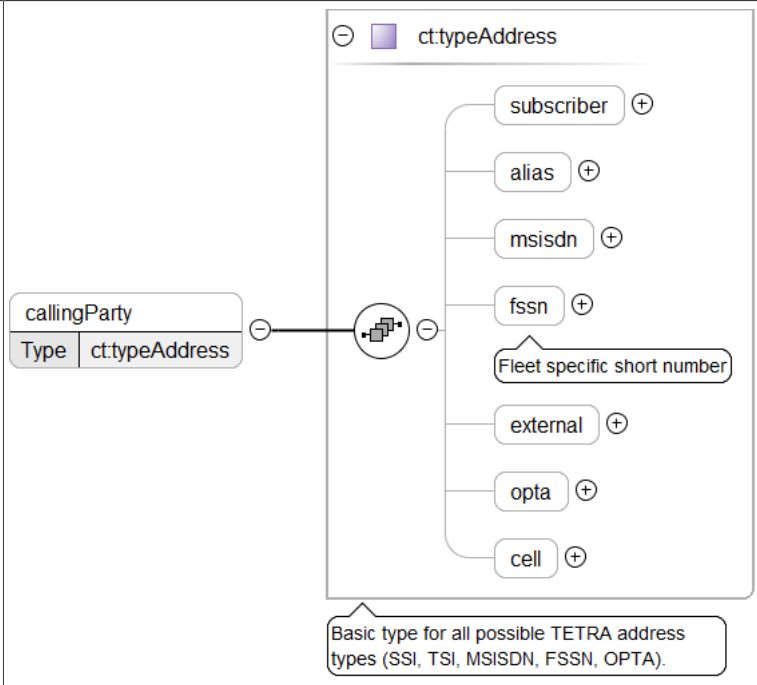
Namespace	DR-GW-Interface/DR-GW-Call
Diagram	
Type	typeActionRequest

Properties	content:	simple	
Facets	enumeration	setup	This method is used to initiate a new call setup. For a call setup to be successful it is required that the resources have been reserved prior this method call.
	enumeration	connect	This method is used to connect an incoming call.
	enumeration	hold	This method requests to put an individual call to hold.
	enumeration	unhold	This method is a request for resuming an individual call from hold.
	enumeration	disconnect	This method is used to disconnect a call.
	enumeration	transfer	This method is used to transfer an individual call to a new recipient.
	enumeration	releasecall	This method is used to release radio subscriber's individual call.
Source	<xs:element name="action" type="ctC:typeActionRequest"/>		

Element Call_Request / attributes

Namespace	DR-GW-Interface/DR-GW-Call
Diagram	
Type	typeCallAttributes
Properties	content: complex minOccurs: 0
Model	hook{0,1} , mode{0,1} , commtype{0,1} , priority{0,1} , encryption{0,1} , ambienceListen{0,1} , req2speak{0,1} , demandPriority{0,1}
Children	ambienceListen, commtype, demandPriority, encryption, hook, mode, priority, req2speak
Instance	<attributes xmlns="DR-GW-Interface/DR-GW-Call" xmlns:ctC="DR-GW-Interface/DR-GW-Call.CommonTypes"> <ctC:hook>{0,1}</ctC:hook> <ctC:mode>{0,1}</ctC:mode> <ctC:commtype>{0,1}</ctC:commtype> <ctC:priority>{0,1}</ctC:priority> <ctC:encryption>{0,1}</ctC:encryption> <ctC:ambienceListen>{0,1}</ctC:ambienceListen> <ctC:req2speak>{0,1}</ctC:req2speak> <ctC:demandPriority>{0,1}</ctC:demandPriority> </attributes>
Source	<xs:element name="attributes" type="ctC:typeCallAttributes" minOccurs="0"/>

Element `Call_Request / callingParty`

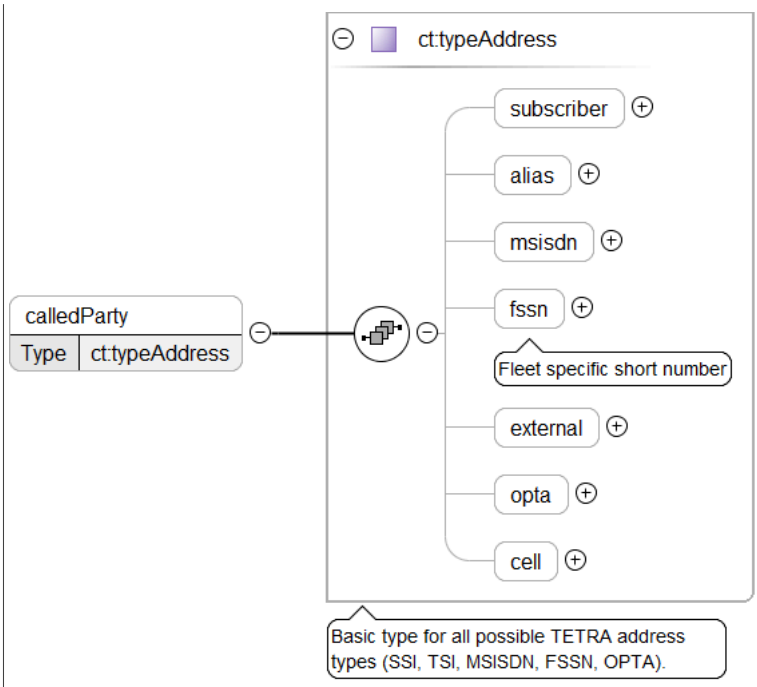
Namespace	DR-GW-Interface/DR-GW-Call
Diagram	
Type	ct:typeAddress
Properties	content: `complex` minOccurs: `0`
Model	ct:subscriber{0,1} , ct:alias{0,1} , ct:msisdn{0,1} , ct:fssn{0,1} , ct:external{0,1} , ct:opta{0,1} , ct:cell{0,1}
Children	ct:alias, ct:cell, ct:external, ct:fssn, ct:msisdn, ct:opta, ct:subscriber

Instance	`<callingParty xmlns="DR-GW-Interface/DR-GW-Call" xmlns:ct="DR-GW-Interface/CommonTypes">`
	`  <ct:subscriber>{0,1}</ct:subscriber>`
	`  <ct:alias>{0,1}</ct:alias>`
	`  <ct:msisdn>{0,1}</ct:msisdn>`
	`  <ct:fssn>{0,1}</ct:fssn>`
	`  <ct:external>{0,1}</ct:external>`
	`  <ct:opta>{0,1}</ct:opta>`
	`  <ct:cell>{0,1}</ct:cell>`
	`</callingParty>`

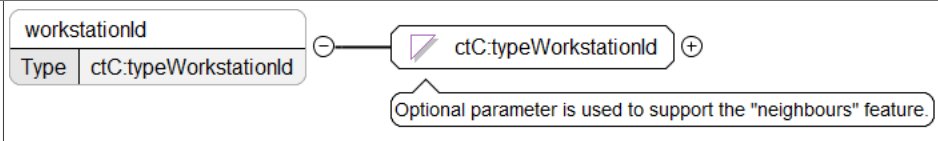
Source	`<xs:element name="callingParty" type="ct:typeAddress" minOccurs="0"/>`

Element `Call_Request / calledParty`

Namespace	DR-GW-Interface/DR-GW-Call

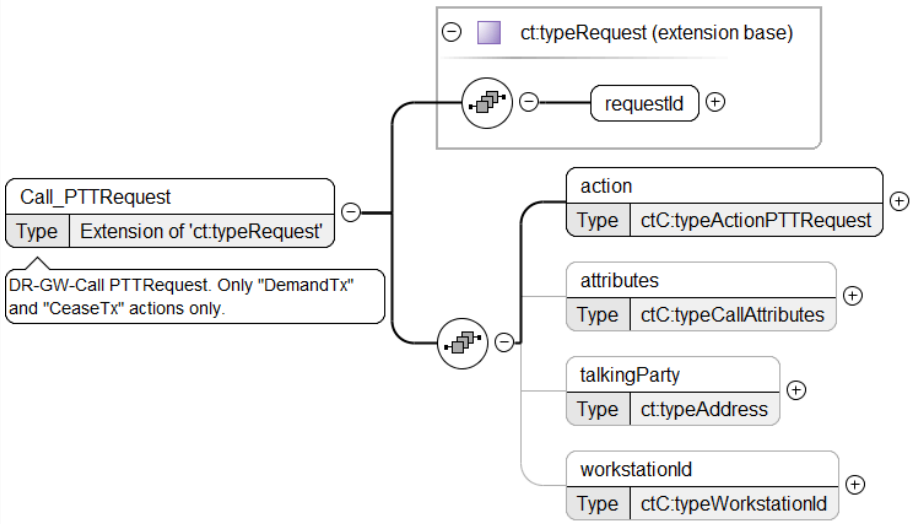
Diagram	
Type	ct:typeAddress
Properties	content: complex minOccurs: 0
Model	ct:subscriber{0,1} , ct:alias{0,1} , ct:msisdn{0,1} , ct:fssn{0,1} , ct:external{0,1} , ct:opta{0,1} , ct:cell{0,1}
Children	ct:alias, ct:cell, ct:external, ct:fssn, ct:msisdn, ct:opta, ct:subscriber
Instance	<calledParty xmlns="DR-GW-Interface/DR-GW-Call" xmlns:ct="DR-GW-Interface/CommonTypes"> <ct:subscriber>{0,1}</ct:subscriber> <ct:alias>{0,1}</ct:alias> <ct:msisdn>{0,1}</ct:msisdn> <ct:fssn>{0,1}</ct:fssn> <ct:external>{0,1}</ct:external> <ct:opta>{0,1}</ct:opta> <ct:cell>{0,1}</ct:cell> </calledParty>
Source	<xs:element name="calledParty" type="ct:typeAddress" minOccurs="0"/>

Element Call_Request / workstationId

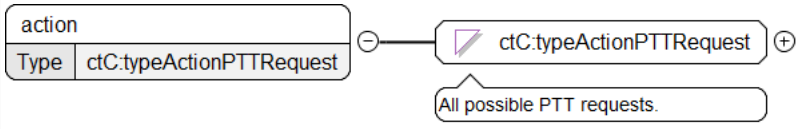
Namespace	DR-GW-Interface/DR-GW-Call
Diagram	
Type	typeWorkstationId
Properties	content: simple minOccurs: 0
Source	<xs:element name="workstationId" type="ctC:typeWorkstationId" minOccurs="0"/>

Element Call_PTTRequest

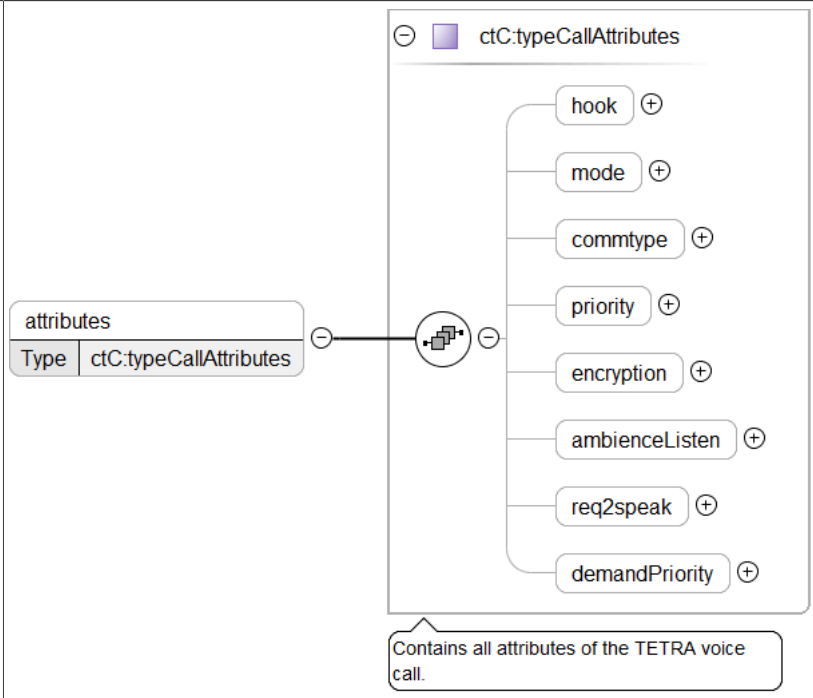
Namespace	DR-GW-Interface/DR-GW-Call
Annotations	DR-GW-Call PTTRequest. Only "DemandTx" and "CeaseTx" actions only.

Diagram	
Type	extension of ct:typeRequest
Type hierarchy	• ct:typeRequest
Properties	content: complex
Model	ct:requestId , action , attributes{0,1} , talkingParty{0,1} , workstationId{0,1}
Children	action, attributes, ct:requestId, talkingParty, workstationId
Instance	<Call_PTTRequest xmlns="DR-GW-Interface/DR-GW-Call" xmlns:ct="DR-GW-Interface/CommonTypes"> <ct:requestId>{1,1}</ct:requestId> <action>{1,1}</action> <attributes>{0,1}</attributes> <talkingParty>{0,1}</talkingParty> <workstationId>{0,1}</workstationId> </Call_PTTRequest>
Source	<xs:element name="Call_PTTRequest"> <xs:annotation> <xs:documentation>DR-GW-Call PTTRequest. Only "DemandTx" and "CeaseTx" actions only.</xs:documentation> </xs:annotation> <xs:complexType> <xs:complexContent> <xs:extension base="ct:typeRequest"> <xs:sequence> <xs:element name="action" type="ctC:typeActionPTTRequest"/> <xs:element name="attributes" type="ctC:typeCallAttributes" minOccurs="0"/> <xs:element name="talkingParty" type="ct:typeAddress" minOccurs="0"/> <xs:element name="workstationId" type="ctC:typeWorkstationId" minOccurs="0"/> </xs:sequence> </xs:extension> </xs:complexContent> </xs:complexType> </xs:element>

Element Call_PTTRequest / action

Namespace	DR-GW-Interface/DR-GW-Call		
Diagram			
Type	typeActionPTTRequest		
Properties	content:	simple	
Facets	enumeration	demandtx	This method can be used to request a speech item for a connected call.
	enumeration	ceasetx	This method is used to inform the system that the speech item is not needed any more.
Source	<xs:element name="action" type="ctC:typeActionPTTRequest"/>		

Element `Call_PTTRequest / attributes`

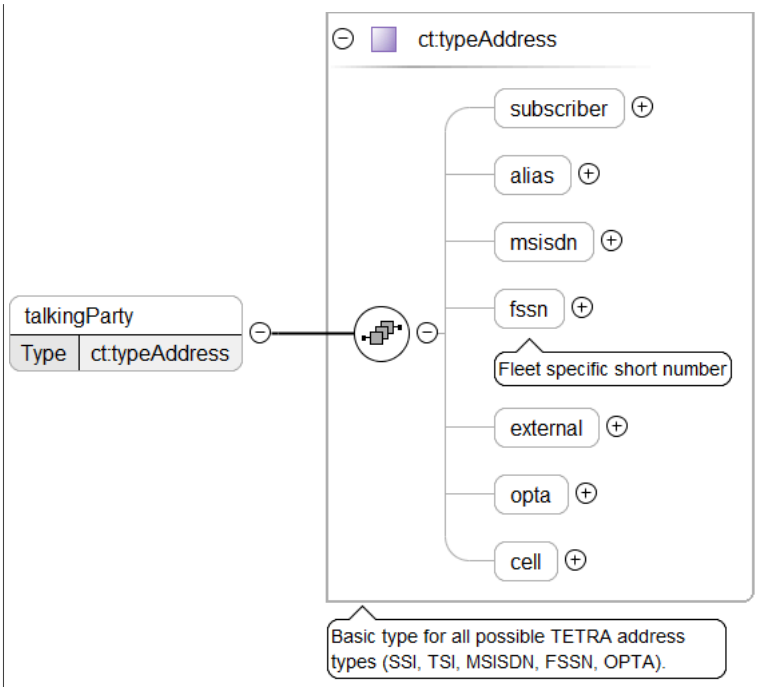
Namespace	DR-GW-Interface/DR-GW-Call
Diagram	
Type	typeCallAttributes
Properties	content: `complex` minOccurs: `0`
Model	hook{0,1} , mode{0,1} , commtype{0,1} , priority{0,1} , encryption{0,1} , ambienceListen{0,1} , req2speak{0,1} , demandPriority{0,1}
Children	ambienceListen, commtype, demandPriority, encryption, hook, mode, priority, req2speak

Instance	`<attributes xmlns="DR-GW-Interface/DR-GW-Call" xmlns:ctC="DR-GW-Interface/DR-GW-Call.CommonTypes">`
	`  <ctC:hook>{0,1}</ctC:hook>`
	`  <ctC:mode>{0,1}</ctC:mode>`
	`  <ctC:commtype>{0,1}</ctC:commtype>`
	`  <ctC:priority>{0,1}</ctC:priority>`
	`  <ctC:encryption>{0,1}</ctC:encryption>`
	`  <ctC:ambienceListen>{0,1}</ctC:ambienceListen>`
	`  <ctC:req2speak>{0,1}</ctC:req2speak>`
	`  <ctC:demandPriority>{0,1}</ctC:demandPriority>`
	`</attributes>`

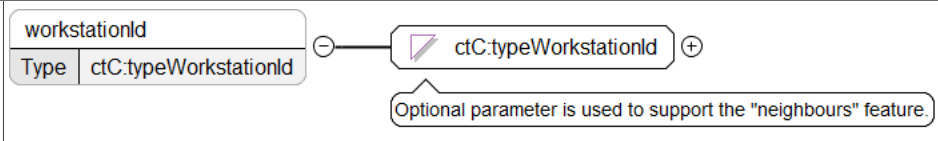
Source	`<xs:element name="attributes" type="ctC:typeCallAttributes" minOccurs="0"/>`

Element `Call_PTTRequest / talkingParty`

Namespace	DR-GW-Interface/DR-GW-Call

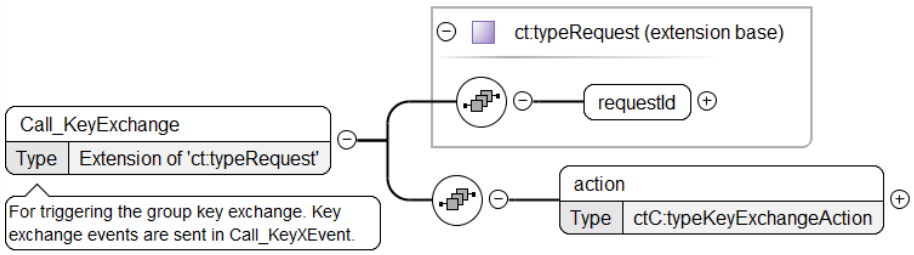
Diagram	
Type	ct:typeAddress
Properties	content: complex minOccurs: 0
Model	ct:subscriber{0,1} , ct:alias{0,1} , ct:msisdn{0,1} , ct:fssn{0,1} , ct:external{0,1} , ct:opta{0,1} , ct:cell{0,1}
Children	ct:alias, ct:cell, ct:external, ct:fssn, ct:msisdn, ct:opta, ct:subscriber
Instance	<talkingParty xmlns="DR-GW-Interface/DR-GW-Call" xmlns:ct="DR-GW-Interface/CommonTypes"> <ct:subscriber>{0,1}</ct:subscriber> <ct:alias>{0,1}</ct:alias> <ct:msisdn>{0,1}</ct:msisdn> <ct:fssn>{0,1}</ct:fssn> <ct:external>{0,1}</ct:external> <ct:opta>{0,1}</ct:opta> <ct:cell>{0,1}</ct:cell> </talkingParty>
Source	<xs:element name="talkingParty" type="ct:typeAddress" minOccurs="0"/>

Element Call_PTTRequest / workstationId

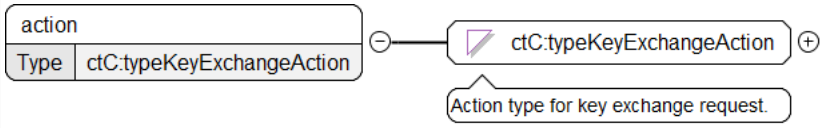
Namespace	DR-GW-Interface/DR-GW-Call
Diagram	
Type	typeWorkstationId
Properties	content: simple minOccurs: 0
Source	<xs:element name="workstationId" type="ctC:typeWorkstationId" minOccurs="0"/>

Element Call_KeyExchange

Namespace	DR-GW-Interface/DR-GW-Call
Annotations	For triggering the group key exchange. Key exchange events are sent in Call_KeyXEvent.

Diagram	
Type	extension of ct:typeRequest
Type hierarchy	• ct:typeRequest
Properties	content: complex
Model	ct:requestId , action
Children	action, ct:requestId
Instance	<Call_KeyExchange xmlns="DR-GW-Interface/DR-GW-Call" xmlns:ct="DR-GW-Interface/CommonTypes"> <ct:requestId>{1,1}</ct:requestId> <action>{1,1}</action> </Call_KeyExchange>
Source	<xs:element name="Call_KeyExchange"> <xs:annotation> <xs:documentation>For triggering the group key exchange. Key exchange events are sent in Call_KeyXEvent.</xs:documentation> </xs:annotation> <xs:complexType> <xs:complexContent> <xs:extension base="ct:typeRequest"> <xs:sequence> <xs:element name="action" type="ctC:typeKeyExchangeAction"/> </xs:sequence> </xs:extension> </xs:complexContent> </xs:complexType> </xs:element>

Element Call_KeyExchange / action

Namespace	DR-GW-Interface/DR-GW-Call
Diagram	
Type	typeKeyExchangeAction
Properties	content: simple
Facets	enumeration start enumeration stop
Source	<xs:element name="action" type="ctC:typeKeyExchangeAction"/>

Namespace: "DR-GW-Interface/CommonTypes"

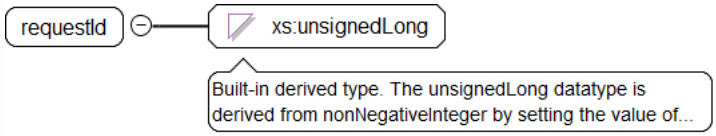
Schema(s)

Imported schema CommonTypes.xsd

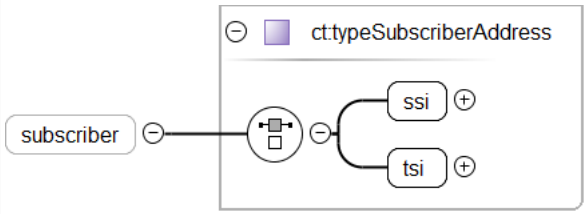
Namespace	DR-GW-Interface/CommonTypes
Annotations	Version 1.1.1
Properties	attribute form default: unqualified element form default: qualified

Element(s)

Element `ct:typeRequest / ct:requestId`

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

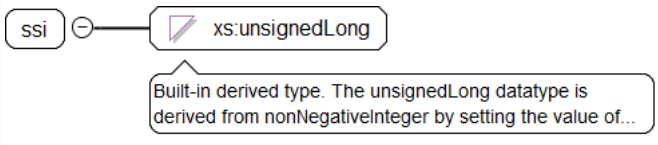
Element `ct:typeAddress / ct:subscriber`

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	`<ct:subscriber xmlns:ct="DR-GW-Interface/CommonTypes">`
	`  <ct:ssi>{1,1}</ct:ssi>`
	`  <ct:tsi>{1,1}</ct:tsi>`
	`</ct:subscriber>`

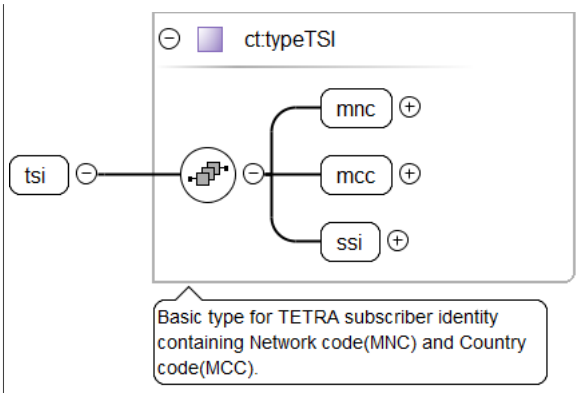
Source	`<xs:element name="subscriber" type="ct:typeSubscriberAddress" minOccurs="0"/>`

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`

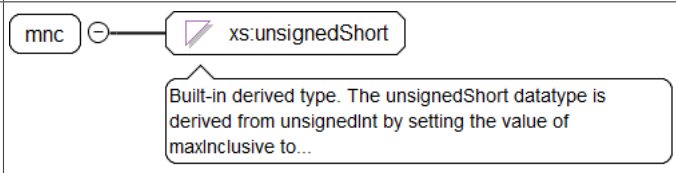
Namespace	DR-GW-Interface/CommonTypes

Diagram	
Type	ct:typeTSI
Properties	content: complex
Model	ct:mnc , ct:mcc , ct:ssi
Children	ct:mcc, ct:mnc, ct:ssi

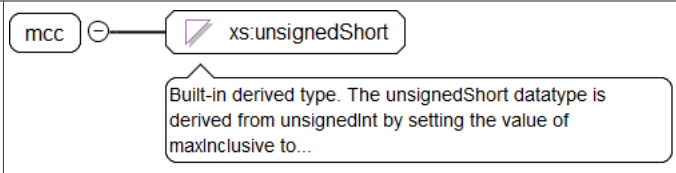
Instance	`<ct:tsi xmlns:ct="DR-GW-Interface/CommonTypes">`
	`<ct:mnc>{1,1}</ct:mnc>`
	`<ct:mcc>{1,1}</ct:mcc>`
	`<ct:ssi>{1,1}</ct:ssi>`
	`</ct:tsi>`

Source	`<xs:element name="tsi" type="ct:typeTSI"/>`

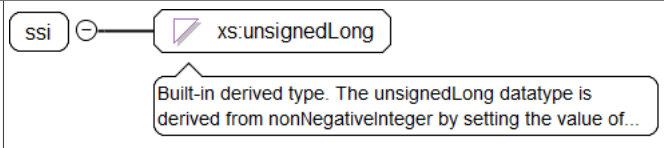
Element ct:typeTSI / ct:mnc

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

Element ct:typeTSI / ct:mcc

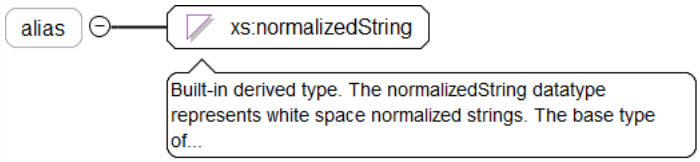
Namespace	DR-GW-Interface/CommonTypes
Diagram	
Type	xs:unsignedShort
Properties	content: simple
Source	`<xs:element name="mcc" type="xs:unsignedShort"/>`

Element ct:typeTSI / ct:ssi

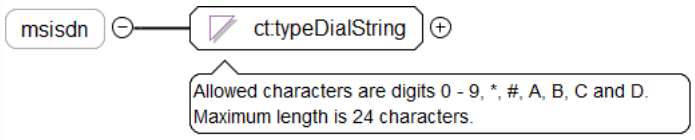
Namespace	DR-GW-Interface/CommonTypes
Diagram	
Type	xs:unsignedLong

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

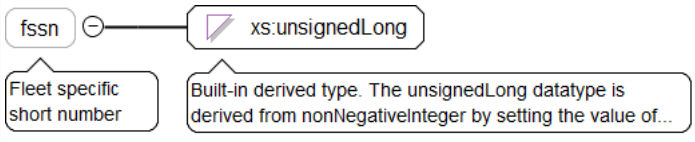
Element ct:typeAddress / ct:alias

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

Element ct:typeAddress / ct:msisdn

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

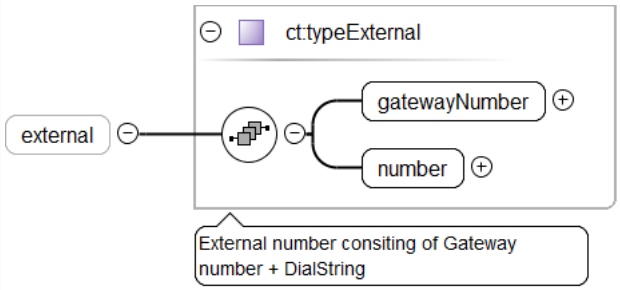
Element ct:typeAddress / ct:fssn

Namespace	DR-GW-Interface/CommonTypes
Annotations	Fleet specific short number
Diagram	
Type	xs:unsignedLong
Properties	content: simple minOccurs: 0

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

Element ct:typeAddress / ct:external

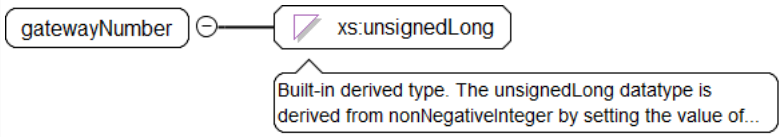
Namespace	DR-GW-Interface/CommonTypes

Diagram	
Type	ct:typeExternal
Properties	content: complex minOccurs: 0
Model	ct:gatewayNumber , ct:number
Children	ct:gatewayNumber, ct:number

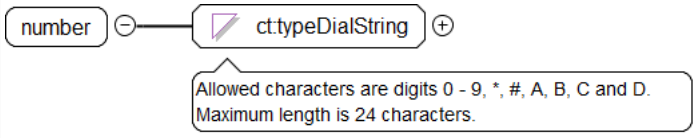
Instance	`<ct:external xmlns:ct="DR-GW-Interface/CommonTypes">`
	`  <ct:gatewayNumber>{1,1}</ct:gatewayNumber>`
	`  <ct:number>{1,1}</ct:number>`
	`</ct:external>`

Source	`<xs:element name="external" type="ct:typeExternal" minOccurs="0"/>`

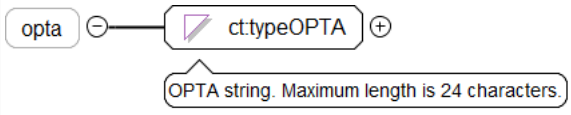
Element ct:typeExternal / ct:gatewayNumber

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

Element ct:typeExternal / ct:number

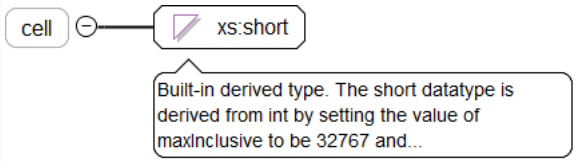
Namespace	DR-GW-Interface/CommonTypes
Diagram	
Type	ct:typeDialString
Properties	content: simple
Facets	maxLength 24
Source	`<xs:element name="number" type="ct:typeDialString"/>`

Element ct:typeAddress / ct:opta

Namespace	DR-GW-Interface/CommonTypes
Diagram	
Type	ct:typeOPTA
Properties	content: simple minOccurs: 0

Facets	maxLength	24
Source	<xs:element name="opta" type="ct:typeOPTA" minOccurs="0"/>	

Element ct:typeAddress / ct:cell

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

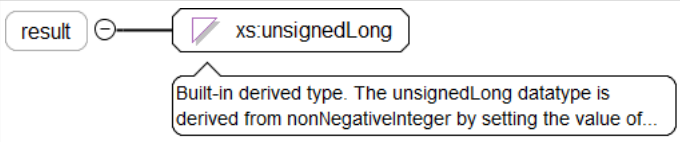
Element ct:typeResult / ct:responseCode

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

Element ct:typeResult / ct:sourceSystem

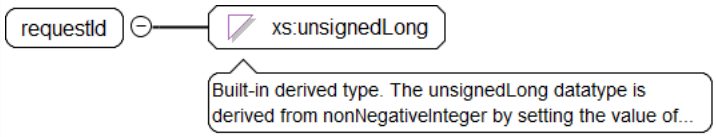
Namespace	DR-GW-Interface/CommonTypes	
Diagram		
Type	ct:typeSourceSystem	
Properties	content:	simple
	minOccurs:	0
Facets	enumeration	DR-GW
	enumeration	TCS-API
	enumeration	TETRA
Source	<xs:element name="sourceSystem" type="ct:typeSourceSystem" minOccurs="0"/>	

Element ct:typeResult / ct:result

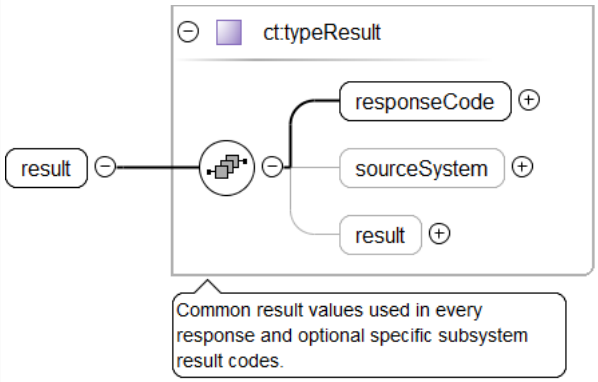
Namespace	DR-GW-Interface/CommonTypes	
Diagram		

Type	xs:unsignedLong
Properties	content: simple minOccurs: 0
Source	`<xs:element name="result" type="xs:unsignedLong" minOccurs="0"/>`

Element ct:typeResponse / ct:requestId

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

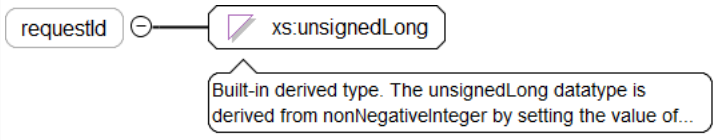
Element ct:typeResponse / ct:result

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

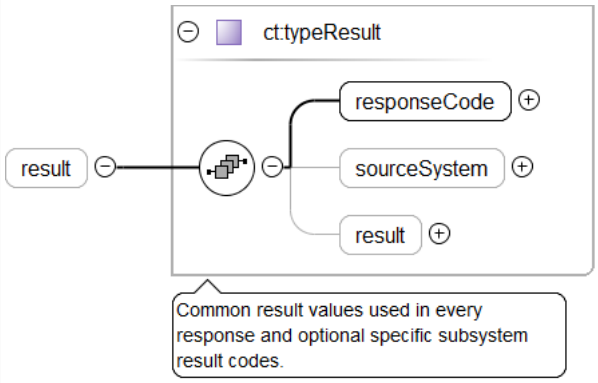
Instance	`<ct:result xmlns:ct="DR-GW-Interface/CommonTypes">`
	`  <ct:responseCode>{1,1}</ct:responseCode>`
	`  <ct:sourceSystem>{0,1}</ct:sourceSystem>`
	`  <ct:result>{0,1}</ct:result>`
	`</ct:result>`

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

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`

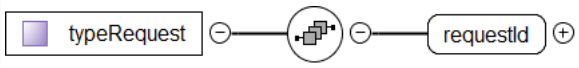
Namespace	DR-GW-Interface/CommonTypes
Diagram	
Type	ct:typeResult
Properties	content: complex minOccurs: 0
Model	ct:responseCode , ct:sourceSystem{0,1} , ct:result{0,1}
Children	ct:responseCode, ct:result, ct:sourceSystem

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"/>`

Complex Type(s)

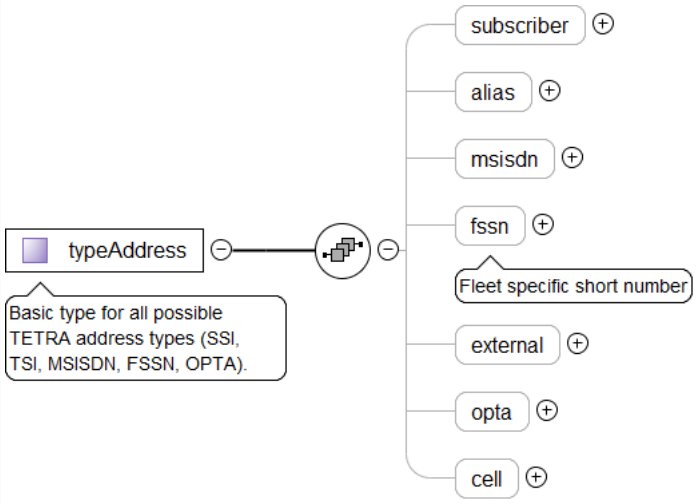
Complex Type `ct:typeRequest`

Namespace	DR-GW-Interface/CommonTypes
Diagram	
Used by	Elements: Call_KeyExchange, Call_PTTRequest, Call_Request, Call_Select
Model	ct:requestId
Children	ct:requestId

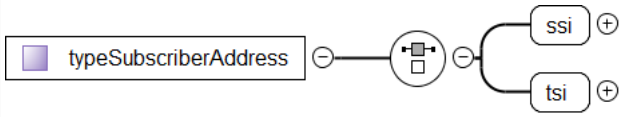
Source	`<xs:complexType name="typeRequest">`
	`  <xs:sequence>`
	`  <xs:element name="requestId" type="xs:unsignedLong"/>`
	`  </xs:sequence>`
	`</xs:complexType>`

Complex Type `ct:typeAddress`

Namespace	DR-GW-Interface/CommonTypes
Annotations	Basic type for all possible TETRA address types (SSI, TSI, MSISDN, FSSN, OPTA).

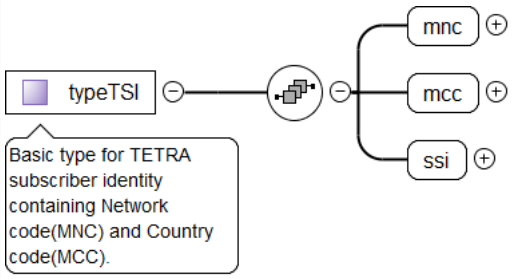
Diagram	
Used by	Elements Call_PTTRequest/talkingParty, Call_Request/calledParty, Call_Request/callingParty, typeSelection/target, typeTxGranted/talkingParty
Model	ct:subscriber{0,1} , ct:alias{0,1} , ct:msisdn{0,1} , ct:fssn{0,1} , ct:external{0,1} , ct:opta{0,1} , ct:cell{0,1}
Children	ct:alias, ct:cell, ct:external, ct:fssn, ct:msisdn, ct:opta, ct:subscriber
Source	<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>

Complex Type `ct:typeSubscriberAddress`

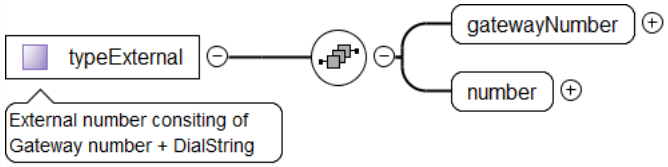
Namespace	DR-GW-Interface/CommonTypes
Annotations	
Diagram	
Used by	Element ct:typeAddress/ct:subscriber
Model	ct:ssi | ct:tsi
Children	ct:ssi, ct:tsi
Source	<xs:complexType name="typeSubscriberAddress"> <xs:annotation> <xs:documentation/> </xs:annotation> <xs:choice> <xs:element name="ssi" type="xs:unsignedLong"/>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

Complex Type `ct:typeTSI`

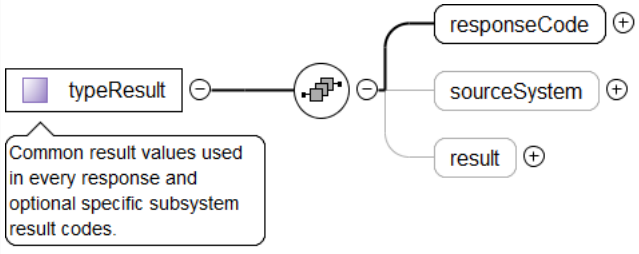
Namespace	DR-GW-Interface/CommonTypes

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	<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>

Complex Type ct:typeExternal

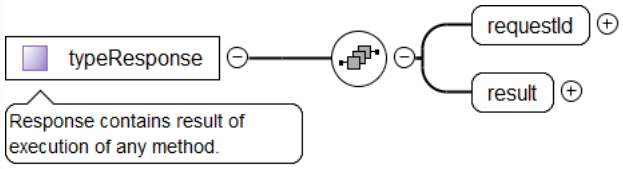
Namespace	DR-GW-Interface/CommonTypes
Annotations	External number consiting of Gateway number + DialString
Diagram	
Used by	Element ct:typeAddress/ct:external
Model	ct:gatewayNumber , ct:number
Children	ct:gatewayNumber, ct:number
Source	<xs:complexType name="typeExternal"> <xs:annotation> <xs:documentation>External number consiting 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>

Complex Type ct:typeResult

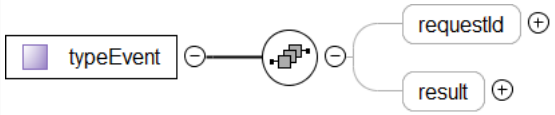
Namespace	DR-GW-Interface/CommonTypes
Annotations	Common result values used in every response and optional specific subsystem result codes.
Diagram	

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	<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>

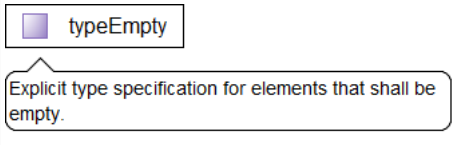
Complex Type ct:typeResponse

Namespace	DR-GW-Interface/CommonTypes
Annotations	Response contains result of execution of any method.
Diagram	
Model	ct:requestId , ct:result
Children	ct:requestId, ct:result
Source	<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>

Complex Type ct:typeEvent

Namespace	DR-GW-Interface/CommonTypes
Diagram	
Model	ct:requestId{0,1} , ct:result{0,1}
Children	ct:requestId, ct:result
Source	<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>

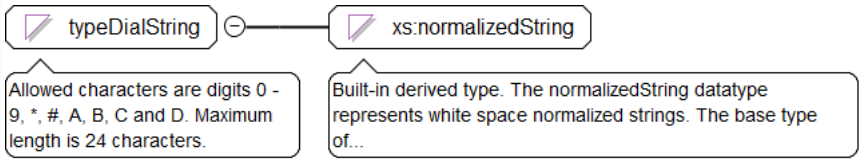
Complex Type ct:typeEmpty

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

```
    <xs:documentation>Explicit type specification for elements that shall be empty.</
xs:documentation>
  </xs:annotation>
</xs:complexType>
```

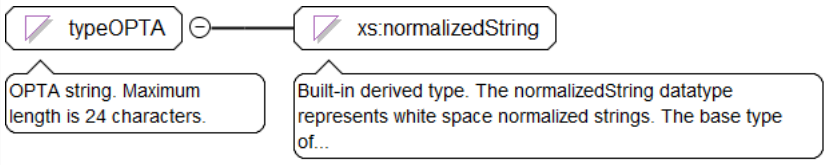
Simple Type(s)

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

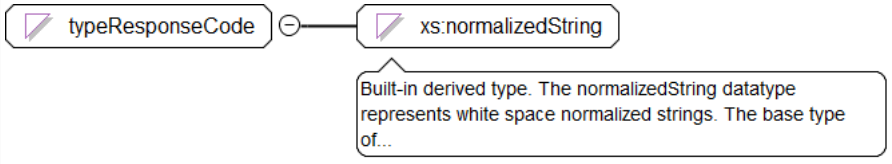
Source	`<xs:simpleType name="typeDialString">`
	`  <xs:annotation>`
	`  <xs:documentation>Allowed characters are digits 0 - 9, *, #, A, B, C and D. Maximum length is 24 characters.</xs:documentation>`
	`  </xs:annotation>`
	`  <xs:restriction base="xs:normalizedString">`
	`  <xs:maxLength value="24"/>`
	`  </xs:restriction>`
	`</xs:simpleType>`

Simple Type `ct:typeOPTA`

Namespace	DR-GW-Interface/CommonTypes
Annotations	OPTA string. Maximum length is 24 characters.
Diagram	
Type	restriction of xs:normalizedString
Facets	maxLength 24
Used by	Element ct:typeAddress/ct:opta

Source	`<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>`

Simple Type `ct:typeResponseCode`

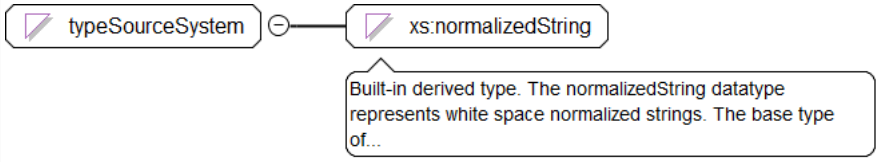
Namespace	DR-GW-Interface/CommonTypes
Diagram	
Type	restriction of xs:normalizedString

Facets	enumeration	success
	enumeration	final_response_pending
	enumeration	error
	enumeration	not_authorized_error
	enumeration	temporary_failure
	enumeration	subscription_failed
Used by	Element	ct:typeResult/ct:responseCode

Source

```
<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>
```

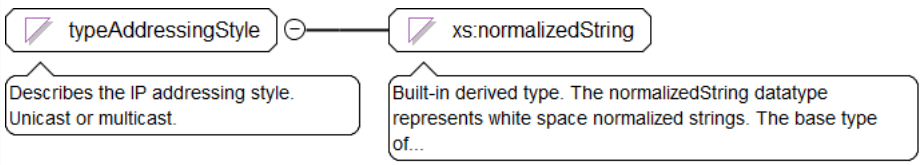
Simple Type `ct:typeSourceSystem`

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

```
<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>
```

Simple Type `ct:typeAddressingStyle`

Namespace	DR-GW-Interface/CommonTypes	
Annotations	Describes the IP addressing style. Unicast or multicast.	
Diagram		
Type	restriction of xs:normalizedString	
Facets	enumeration	ucast
	enumeration	mcast

Source

```
<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>
```

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

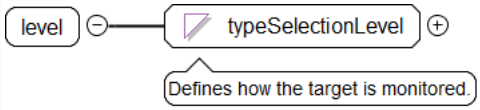
Schema(s)

Imported schema `DR-GW-Call.CommonTypes.xsd`

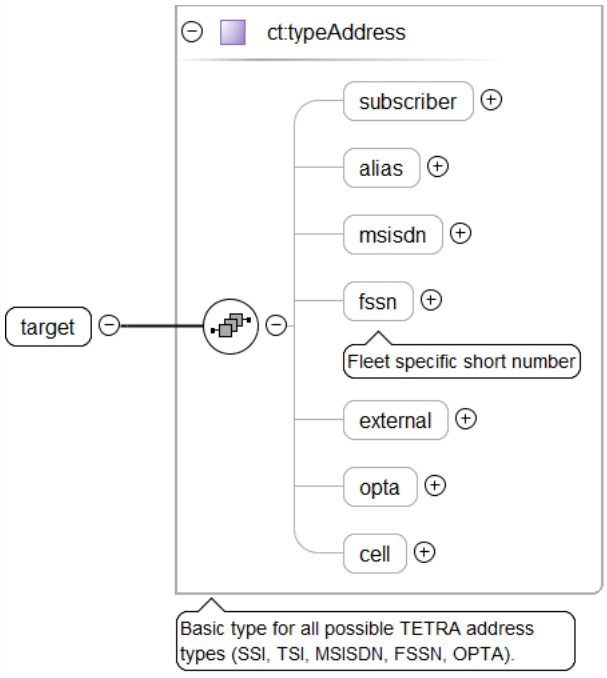
Namespace	DR-GW-Interface/DR-GW-Call.CommonTypes
Annotations	`Version 1.1.1`
Properties	attribute form default: `unqualified` element form default: `qualified`

Element(s)

Element `typeSelection / level`

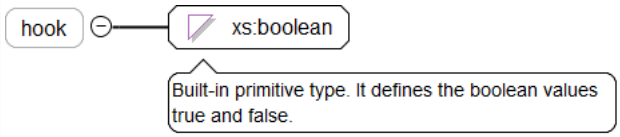
Namespace	DR-GW-Interface/DR-GW-Call.CommonTypes		
Diagram			
Type	typeSelectionLevel		
Properties	content:	`simple`	
Facets	enumeration	`no`	`No selection. Used to remove selection.`
	enumeration	`event`	`Event monitoring.`
	enumeration	`audio`	`Audio monitoring.`
	enumeration	`use`	`Selection level use.`
	enumeration	`a_use`	`Selection level active use.`
Source	`<xs:element name="level" type="typeSelectionLevel"/>`		

Element `typeSelection / target`

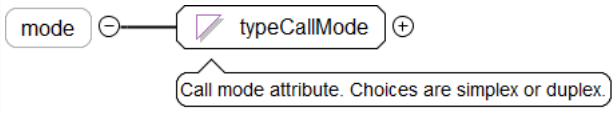
Namespace	DR-GW-Interface/DR-GW-Call.CommonTypes
Diagram	
Type	ct:typeAddress
Properties	content: `complex`
Model	ct:subscriber{0,1} , ct:alias{0,1} , ct:msisdn{0,1} , ct:fssn{0,1} , ct:external{0,1} , ct:opta{0,1} , ct:cell{0,1}

Children	ct:alias, ct:cell, ct:external, ct:fssn, ct:msisdn, ct:opta, ct:subscriber
Instance	<target xmlns="DR-GW-Interface/DR-GW-Call.CommonTypes" xmlns:ct="DR-GW-Interface/CommonTypes"> <ct:subscriber>{0,1}</ct:subscriber> <ct:alias>{0,1}</ct:alias> <ct:msisdn>{0,1}</ct:msisdn> <ct:fssn>{0,1}</ct:fssn> <ct:external>{0,1}</ct:external> <ct:opta>{0,1}</ct:opta> <ct:cell>{0,1}</ct:cell> </target>
Source	<xs:element name="target" type="ct:typeAddress"/>

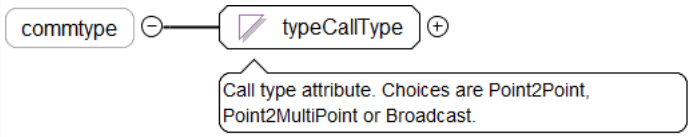
Element typeCallAttributes / hook

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

Element typeCallAttributes / mode

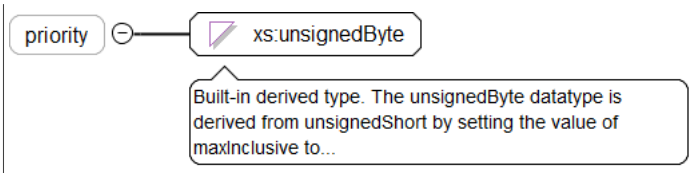
Namespace	DR-GW-Interface/DR-GW-Call.CommonTypes
Diagram	
Type	typeCallMode
Properties	content: simple minOccurs: 0
Facets	enumeration simplex enumeration duplex
Source	<xs:element name="mode" type="typeCallMode" minOccurs="0"/>

Element typeCallAttributes / commtype

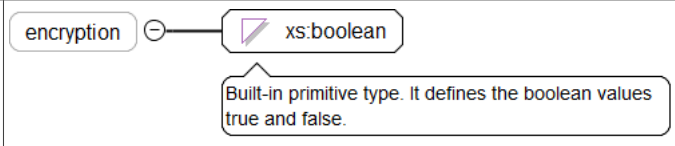
Namespace	DR-GW-Interface/DR-GW-Call.CommonTypes
Diagram	
Type	typeCallType
Properties	content: simple minOccurs: 0
Facets	enumeration p2p enumeration p2mp enumeration bcast
Source	<xs:element name="commtype" type="typeCallType" minOccurs="0"/>

Element typeCallAttributes / priority

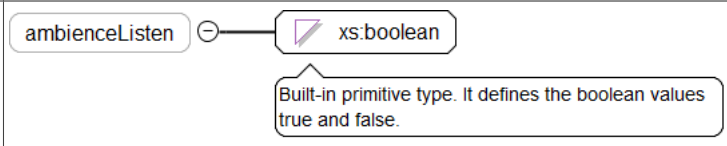
Namespace	DR-GW-Interface/DR-GW-Call.CommonTypes

Diagram	
Type	xs:unsignedByte
Properties	content: simple minOccurs: 0 default: 1
Source	`<xs:element name="priority" type="xs:unsignedByte" default="1" minOccurs="0"/>`

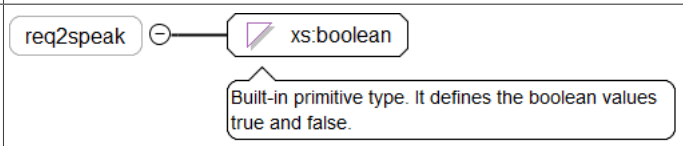
Element typeCallAttributes / encryption

Namespace	DR-GW-Interface/DR-GW-Call.CommonTypes
Diagram	
Type	xs:boolean
Properties	content: simple minOccurs: 0 default: true
Source	`<xs:element name="encryption" type="xs:boolean" default="true" minOccurs="0"/>`

Element typeCallAttributes / ambienceListen

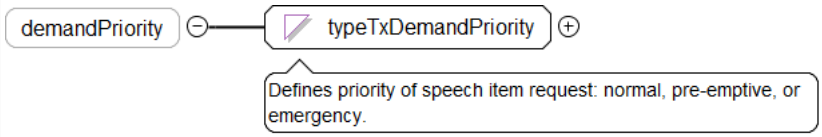
Namespace	DR-GW-Interface/DR-GW-Call.CommonTypes
Diagram	
Type	xs:boolean
Properties	content: simple minOccurs: 0 default: 0
Source	`<xs:element name="ambienceListen" type="xs:boolean" default="0" minOccurs="0"/>`

Element typeCallAttributes / req2speak

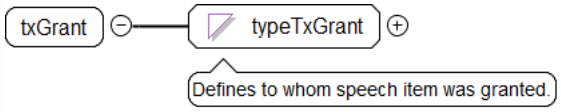
Namespace	DR-GW-Interface/DR-GW-Call.CommonTypes
Diagram	
Type	xs:boolean
Properties	content: simple minOccurs: 0 default: 1
Source	`<xs:element name="req2speak" type="xs:boolean" default="1" minOccurs="0"/>`

Element typeCallAttributes / demandPriority

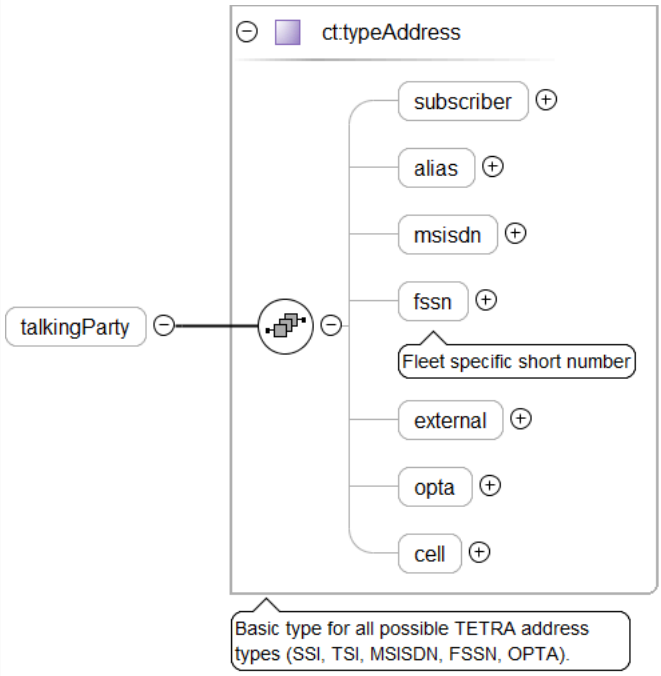
Namespace	DR-GW-Interface/DR-GW-Call.CommonTypes

Diagram	
Type	typeTxDemandPriority
Properties	content: simple minOccurs: 0 default: normal
Facets	enumeration normal enumeration preemptive enumeration emergency
Source	`<xs:element name="demandPriority" type="typeTxDemandPriority" default="normal" minOccurs="0"/>`

Element typeTxGranted / txGrant

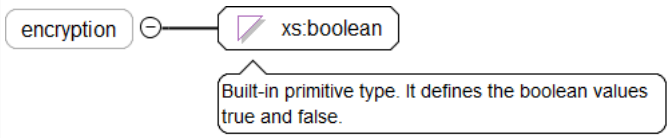
Namespace	DR-GW-Interface/DR-GW-Call.CommonTypes
Diagram	
Type	typeTxGrant
Properties	content: simple
Facets	enumeration granted enumeration notGranted enumeration queued enumeration granted2another
Source	`<xs:element name="txGrant" type="typeTxGrant"/>`

Element typeTxGranted / talkingParty

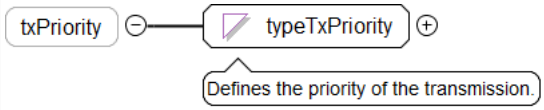
Namespace	DR-GW-Interface/DR-GW-Call.CommonTypes
Diagram	
Type	ct:typeAddress

Properties	content:	complex
	minOccurs:	0
Model	ct:subscriber{0,1} , ct:alias{0,1} , ct:msisdn{0,1} , ct:fssn{0,1} , ct:external{0,1} , ct:opta{0,1} , ct:cell{0,1}	
Children	ct:alias, ct:cell, ct:external, ct:fssn, ct:msisdn, ct:opta, ct:subscriber	
Instance	<talkingParty xmlns="DR-GW-Interface/DR-GW-Call.CommonTypes" xmlns:ct="DR-GW-Interface/CommonTypes"> <ct:subscriber>{0,1}</ct:subscriber> <ct:alias>{0,1}</ct:alias> <ct:msisdn>{0,1}</ct:msisdn> <ct:fssn>{0,1}</ct:fssn> <ct:external>{0,1}</ct:external> <ct:opta>{0,1}</ct:opta> <ct:cell>{0,1}</ct:cell> </talkingParty>	
Source	<xs:element name="talkingParty" type="ct:typeAddress" minOccurs="0"/>	

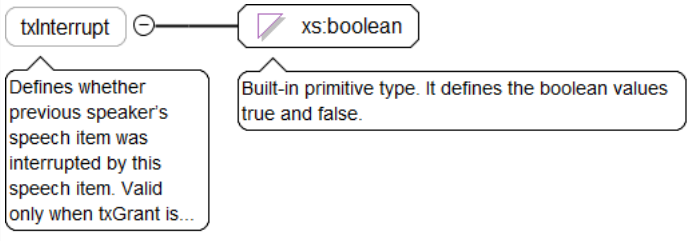
Element typeTxGranted / encryption

Namespace	DR-GW-Interface/DR-GW-Call.CommonTypes
Diagram	
Type	xs:boolean
Properties	content: simple minOccurs: 0 default: true
Source	<xs:element name="encryption" type="xs:boolean" default="true" minOccurs="0"/>

Element typeTxGranted / txPriority

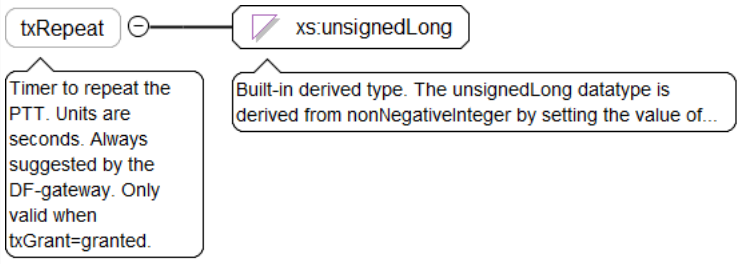
Namespace	DR-GW-Interface/DR-GW-Call.CommonTypes
Diagram	
Type	typeTxPriority
Properties	content: simple minOccurs: 0 default: normal
Facets	enumeration: normal enumeration: emergency
Source	<xs:element name="txPriority" type="typeTxPriority" minOccurs="0" default="normal"/>

Element typeTxGranted / txInterrupt

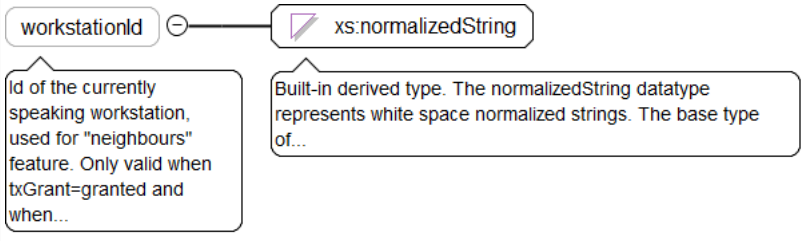
Namespace	DR-GW-Interface/DR-GW-Call.CommonTypes
Annotations	Defines whether previous speaker's speech item was interrupted by this speech item. Valid only when txGrant is granted2another.
Diagram	
Type	xs:boolean

Properties	content:	simple
	minOccurs:	0
	default:	false
Source	<xs:element name="txInterrupt" type="xs:boolean" default="false" minOccurs="0"> <xs:annotation> <xs:documentation>Defines whether previous speaker's speech item was interrupted by this speech item. Valid only when txGrant is granted2another.</xs:documentation> </xs:annotation> </xs:element>	

Element typeTxGranted / txRepeat

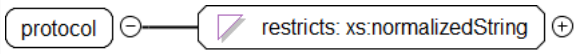
Namespace	DR-GW-Interface/DR-GW-Call.CommonTypes
Annotations	Timer to repeat the PTT. Units are seconds. Always suggested by the DF-gateway. Only valid when txGrant=granted.
Diagram	
Type	xs:unsignedLong
Properties	content: simple minOccurs: 0 default: 0
Source	<xs:element name="txRepeat" type="xs:unsignedLong" minOccurs="0" default="0"> <xs:annotation> <xs:documentation>Timer to repeat the PTT. Units are seconds. Always suggested by the DF-gateway. Only valid when txGrant=granted.</xs:documentation> </xs:annotation> </xs:element>

Element typeTxGranted / workstationId

Namespace	DR-GW-Interface/DR-GW-Call.CommonTypes
Annotations	Id of the currently speaking workstation, used for "neighbours" feature. Only valid when txGrant=granted and when supplied by the DF-client in PTT request.
Diagram	
Type	xs:normalizedString
Properties	content: simple minOccurs: 0
Source	<xs:element name="workstationId" type="xs:normalizedString" minOccurs="0"> <xs:annotation> <xs:documentation>Id of the currently speaking workstation, used for "neighbours" feature. Only valid when txGrant=granted and when supplied by the DF-client in PTT request.</xs:documentation> </xs:annotation> </xs:element>

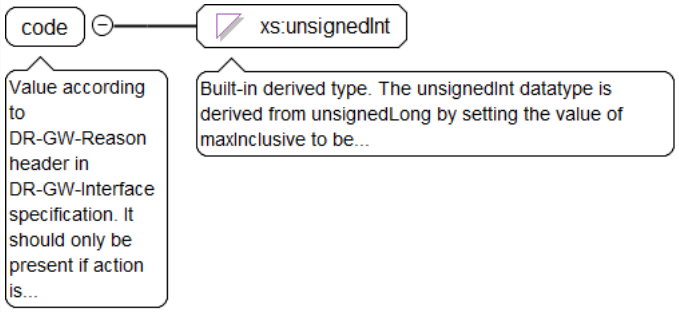
Element typeDisconnectCause / protocol

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

Diagram	
Type	restriction of xs:normalizedString
Properties	content: simple
Facets	enumeration DR-GW enumeration TCS-API

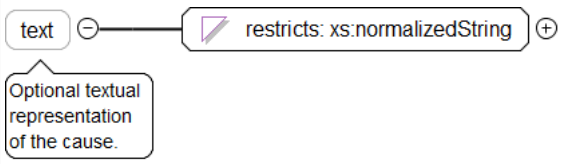
Source	`<xs:element name="protocol">`
	`  <xs:simpleType>`
	`  <xs:restriction base="xs:normalizedString">`
	`  <xs:enumeration value="DR-GW"/>`
	`  <xs:enumeration value="TCS-API"/>`
	`  </xs:restriction>`
	`  </xs:simpleType>`
	`</xs:element>`

Element typeDisconnectCause / code

Namespace	DR-GW-Interface/DR-GW-Call.CommonTypes
Annotations	Value according to DR-GW-Reason header in DR-GW-Interface specification. It should only be present if action is "disconnected" and holds the reason for call disconnection.
Diagram	
Type	xs:unsignedInt
Properties	content: simple

Source	`<xs:element name="code" type="xs:unsignedInt">`
	`  <xs:annotation>`
	`  <xs:documentation>Value according to DR-GW-Reason header in DR-GW-Interface specification. It should only be present if action is "disconnected" and holds the reason for call disconnection.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

Element typeDisconnectCause / text

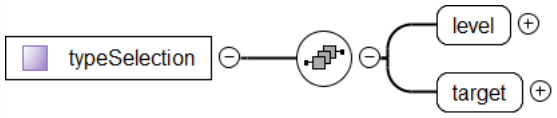
Namespace	DR-GW-Interface/DR-GW-Call.CommonTypes
Annotations	Optional textual representation of the cause.
Diagram	
Type	restriction of xs:normalizedString
Properties	content: simple minOccurs: 0
Facets	maxLength 80

Source	`<xs:element name="text" minOccurs="0">`
	`  <xs:annotation>`
	`  <xs:documentation>Optional textual representation of the cause.</xs:documentation>`
	`  </xs:annotation>`
	`  <xs:simpleType>`
	`  <xs:restriction base="xs:normalizedString">`

```
      <xs:maxLength value="80"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
```

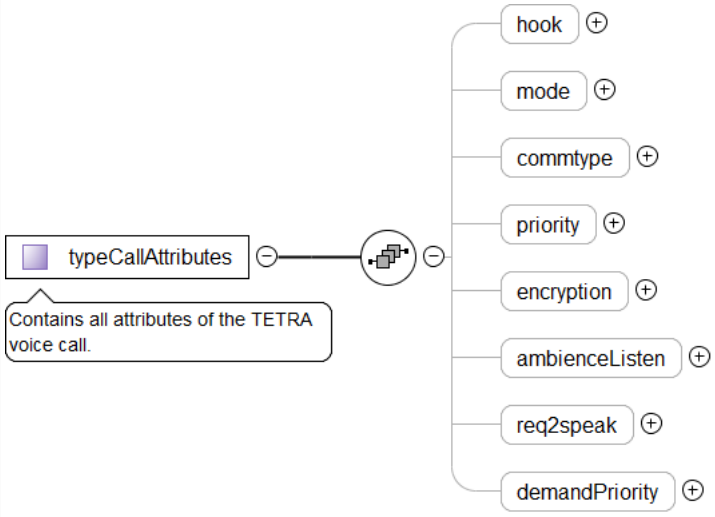
Complex Type(s)

Complex Type `typeSelection`

Namespace	DR-GW-Interface/DR-GW-Call.CommonTypes
Annotations	
Diagram	
Used by	Element Call_Select/sel
Model	level , target
Children	level, target

Source	`<xs:complexType name="typeSelection">`
	`  <xs:annotation>`
	`  <xs:documentation/>`
	`  </xs:annotation>`
	`  <xs:sequence>`
	`  <xs:element name="level" type="typeSelectionLevel"/>`
	`  <xs:element name="target" type="ct:typeAddress"/>`
	`  </xs:sequence>`
	`</xs:complexType>`

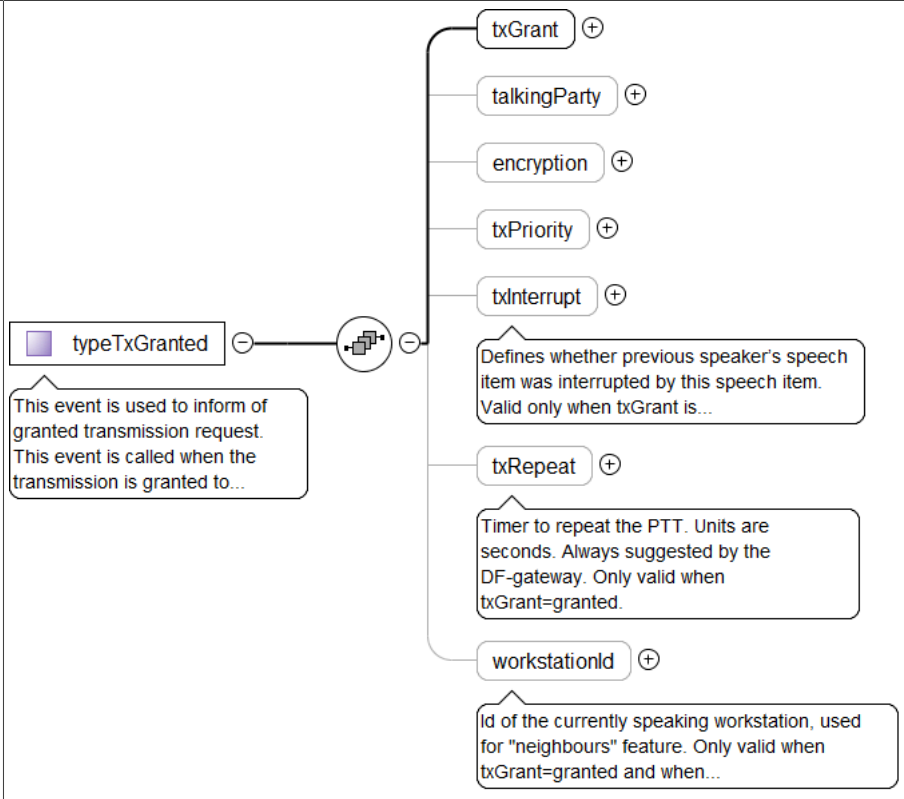
Complex Type `typeCallAttributes`

Namespace	DR-GW-Interface/DR-GW-Call.CommonTypes
Annotations	`Contains all attributes of the TETRA voice call.`
Diagram	
Used by	Elements Call_PTTRequest/attributes, Call_Request/attributes
Model	hook{0,1} , mode{0,1} , commtype{0,1} , priority{0,1} , encryption{0,1} , ambienceListen{0,1} , req2speak{0,1} , demandPriority{0,1}
Children	ambienceListen, commtype, demandPriority, encryption, hook, mode, priority, req2speak

Source	`<xs:complexType name="typeCallAttributes">`
	`  <xs:annotation>`
	`  <xs:documentation>Contains all attributes of the TETRA voice call.</xs:documentation>`
	`  </xs:annotation>`
	`  <xs:sequence>`
	`  <xs:element name="hook" type="xs:boolean" minOccurs="0"/>`
	`  <xs:element name="mode" type="typeCallMode" minOccurs="0"/>`
	`  <xs:element name="commtype" type="typeCallType" minOccurs="0"/>`
	`  <xs:element name="priority" type="xs:unsignedByte" default="1" minOccurs="0"/>`

```
    <xs:element name="encryption" type="xs:boolean" default="true" minOccurs="0"/>
    <xs:element name="ambienceListen" type="xs:boolean" default="0" minOccurs="0"/>
    <xs:element name="req2speak" type="xs:boolean" default="1" minOccurs="0"/>
    <xs:element name="demandPriority" type="typeTxDemandPriority" default="normal" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
```

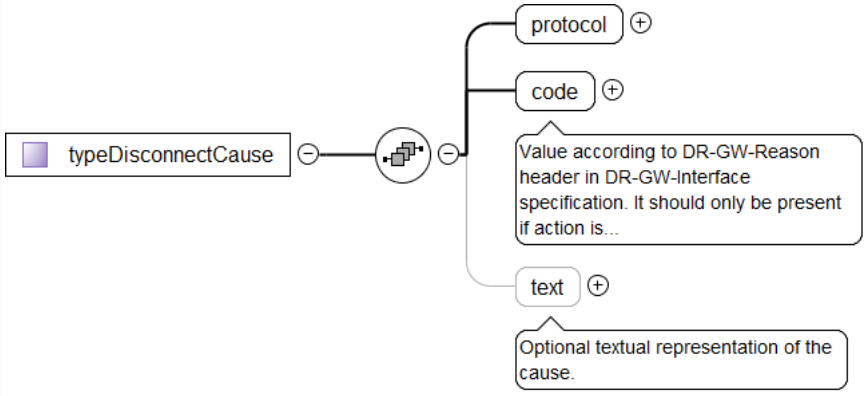
Complex Type `typeTxGranted`

Namespace	DR-GW-Interface/DR-GW-Call.CommonTypes
Annotations	This event is used to inform of granted transmission request. This event is called when the transmission is granted to client, to another user or the request has been queued.
Diagram	
Model	txGrant , talkingParty{0,1} , encryption{0,1} , txPriority{0,1} , txInterrupt{0,1} , txRepeat{0,1} , workstationId{0,1}
Children	encryption, talkingParty, txGrant, txInterrupt, txPriority, txRepeat, workstationId
Source	

```
<xs:complexType name="typeTxGranted">
  <xs:annotation>
    <xs:documentation>This event is used to inform of granted transmission request. This event
 is called when the transmission is granted to client, to another user or the request has been
 queued.</xs:documentation>
  </xs:annotation>
  <xs:sequence>
    <xs:element name="txGrant" type="typeTxGrant"/>
    <xs:element name="talkingParty" type="ct:typeAddress" minOccurs="0"/>
    <xs:element name="encryption" type="xs:boolean" default="true" minOccurs="0"/>
    <xs:element name="txPriority" type="typeTxPriority" minOccurs="0" default="normal"/>
    <xs:element name="txInterrupt" type="xs:boolean" default="false" minOccurs="0">
      <xs:annotation>
        <xs:documentation>Defines whether previous speaker’s speech item was interrupted by this
 speech item. Valid only when txGrant is granted2another.</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="txRepeat" type="xs:unsignedLong" minOccurs="0" default="0">
      <xs:annotation>
        <xs:documentation>Timer to repeat the PTT. Units are seconds. Always suggested by the DF-
gateway. Only valid when txGrant=granted.</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="workstationId" type="xs:normalizedString" minOccurs="0">
      <xs:annotation>
        <xs:documentation>Id of the currently speaking workstation, used for "neighbours"
 feature. Only valid when txGrant=granted and when supplied by the DF-client in PTT request.</
xs:documentation>
```

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

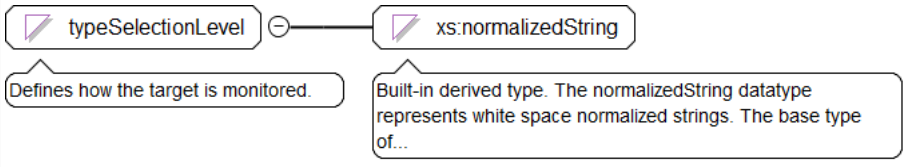
Complex Type `typeDisconnectCause`

Namespace	DR-GW-Interface/DR-GW-Call.CommonTypes
Diagram	
Model	protocol , code , text{0,1}
Children	code, protocol, text

Source	`<xs:complexType name="typeDisconnectCause">`
	`  <xs:sequence>`
	`    <xs:element name="protocol">`
	`      <xs:simpleType>`
	`        <xs:restriction base="xs:normalizedString">`
	`          <xs:enumeration value="DR-GW"/>`
	`          <xs:enumeration value="TCS-API"/>`
	`        </xs:restriction>`
	`      </xs:simpleType>`
	`    </xs:element>`
	`    <xs:element name="code" type="xs:unsignedInt">`
	`      <xs:annotation>`
	`        <xs:documentation>Value according to DR-GW-Reason header in DR-GW-Interface specification.`
	` It should only be present if action is "disconnected" and holds the reason for call`
	` disconnection.</xs:documentation>`
	`      </xs:annotation>`
	`    </xs:element>`
	`    <xs:element name="text" minOccurs="0">`
	`      <xs:annotation>`
	`        <xs:documentation>Optional textual representation of the cause.</xs:documentation>`
	`      </xs:annotation>`
	`      <xs:simpleType>`
	`        <xs:restriction base="xs:normalizedString">`
	`          <xs:maxLength value="80"/>`
	`        </xs:restriction>`
	`      </xs:simpleType>`
	`    </xs:element>`
	`  </xs:sequence>`
	`</xs:complexType>`

Simple Type(s)

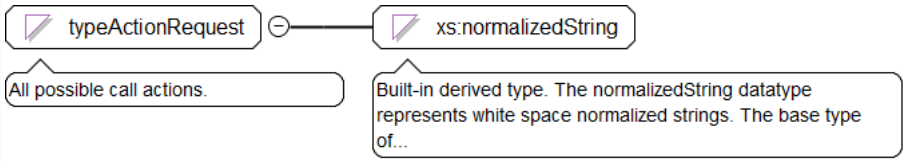
Simple Type `typeSelectionLevel`

Namespace	DR-GW-Interface/DR-GW-Call.CommonTypes		
Annotations	Defines how the target is monitored.		
Diagram			
Type	restriction of xs:normalizedString		
Facets	enumeration	no	No selection. Used to remove selection.
	enumeration	event	Event monitoring.

	enumeration	audio	Audio monitoring.
	enumeration	use	Selection level use.
	enumeration	a_use	Selection level active use.
Used by	Element	typeSelection/level	
Source			

```
<xs:simpleType name="typeSelectionLevel">
  <xs:annotation>
    <xs:documentation>Defines how the target is monitored.</xs:documentation>
  </xs:annotation>
  <xs:restriction base="xs:normalizedString">
    <xs:enumeration value="no">
      <xs:annotation>
        <xs:documentation>No selection. Used to remove selection.</xs:documentation>
      </xs:annotation>
    </xs:enumeration>
    <xs:enumeration value="event">
      <xs:annotation>
        <xs:documentation>Event monitoring.</xs:documentation>
      </xs:annotation>
    </xs:enumeration>
    <xs:enumeration value="audio">
      <xs:annotation>
        <xs:documentation>Audio monitoring.</xs:documentation>
      </xs:annotation>
    </xs:enumeration>
    <xs:enumeration value="use">
      <xs:annotation>
        <xs:documentation>Selection level use.</xs:documentation>
      </xs:annotation>
    </xs:enumeration>
    <xs:enumeration value="a_use">
      <xs:annotation>
        <xs:documentation>Selection level active use.</xs:documentation>
      </xs:annotation>
    </xs:enumeration>
  </xs:restriction>
</xs:simpleType>
```

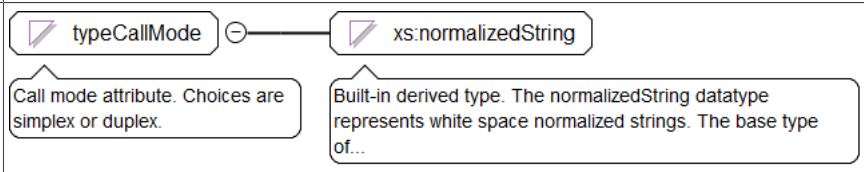
Simple Type `typeActionRequest`

Namespace	DR-GW-Interface/DR-GW-Call.CommonTypes		
Annotations	All possible call actions.		
Diagram			
Type	restriction of xs:normalizedString		
Facets	enumeration	setup	This method is used to initiate a new call setup. For a call setup to be successful it is required that the resources have been reserved prior this method call.
	enumeration	connect	This method is used to connect an incoming call.
	enumeration	hold	This method requests to put an individual call to hold.
	enumeration	unhold	This method is a request for resuming an individual call from hold.
	enumeration	disconnect	This method is used to disconnect a call.
	enumeration	transfer	This method is used to transfer an individual call to a new recipient.
	enumeration	releasecall	This method is used to release radio subscriber's individual call.
Used by	Element	Call_Request/action	
Source			


```
<xs:simpleType name="typeActionRequest">
  <xs:annotation>
    <xs:documentation>All possible call actions.</xs:documentation>
  </xs:annotation>
```

```
  <xs:restriction base="xs:normalizedString">
    <xs:enumeration value="setup">
      <xs:annotation>
        <xs:documentation>This method is used to initiate a new call setup. For a call setup to
 be successful it is required that the resources have been reserved prior this method call.</
xs:documentation>
      </xs:annotation>
    </xs:enumeration>
    <xs:enumeration value="connect">
      <xs:annotation>
        <xs:documentation>This method is used to connect an incoming call.</xs:documentation>
      </xs:annotation>
    </xs:enumeration>
    <xs:enumeration value="hold">
      <xs:annotation>
        <xs:documentation>This method requests to put an individual call to hold.</xs:documentation>
      </xs:annotation>
    </xs:enumeration>
    <xs:enumeration value="unhold">
      <xs:annotation>
        <xs:documentation>This method is a request for resuming an individual call from hold.</
xs:documentation>
      </xs:annotation>
    </xs:enumeration>
    <xs:enumeration value="disconnect">
      <xs:annotation>
        <xs:documentation>This method is used to disconnect a call.</xs:documentation>
      </xs:annotation>
    </xs:enumeration>
    <xs:enumeration value="transfer">
      <xs:annotation>
        <xs:documentation>This method is used to transfer an individual call to a new recipient.</
xs:documentation>
      </xs:annotation>
    </xs:enumeration>
    <xs:enumeration value="releasecall">
      <xs:annotation>
        <xs:documentation>This method is used to release radio subscriber's individual call.</
xs:documentation>
      </xs:annotation>
    </xs:enumeration>
  </xs:restriction>
</xs:simpleType>
```

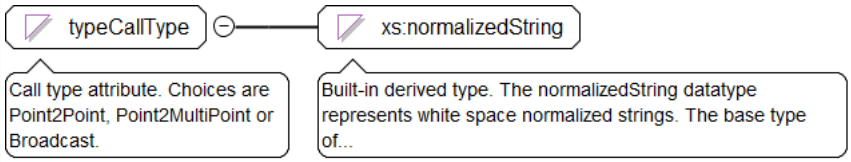
Simple Type `typeCallMode`

Namespace	DR-GW-Interface/DR-GW-Call.CommonTypes
Annotations	`Call mode attribute. Choices are simplex or duplex.`
Diagram	
Type	restriction of xs:normalizedString
Facets	enumeration `simplex` enumeration `duplex`
Used by	Element typeCallAttributes/mode

Source	<pre><xs:simpleType name="typeCallMode">
	  <xs:annotation>
	    <xs:documentation>Call mode attribute. Choices are simplex or duplex.</xs:documentation>
	  </xs:annotation>
	  <xs:restriction base="xs:normalizedString">
	    <xs:enumeration value="simplex"/>
	    <xs:enumeration value="duplex"/>
	  </xs:restriction>
	</xs:simpleType></pre>

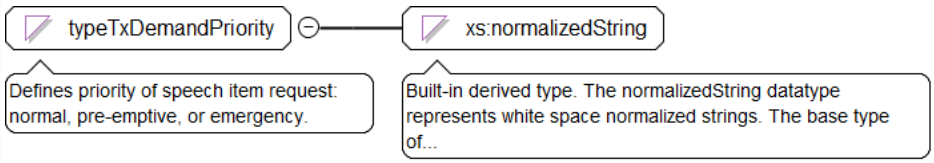
Simple Type `typeCallType`

Namespace	DR-GW-Interface/DR-GW-Call.CommonTypes
Annotations	`Call type attribute. Choices are Point2Point, Point2MultiPoint or Broadcast.`

Diagram	
Type	restriction of xs:normalizedString
Facets	enumeration `p2p` enumeration `p2mp` enumeration `bcast`
Used by	Element typeCallAttributes/commtype

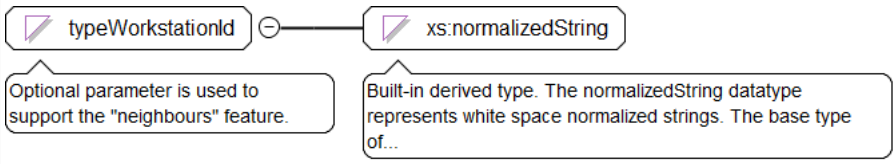
Source	`<xs:simpleType name="typeCallType">`
	`  <xs:annotation>`
	`  <xs:documentation>Call type attribute. Choices are Point2Point, Point2MultiPoint or Broadcast.</xs:documentation>`
	`  </xs:annotation>`
	`  <xs:restriction base="xs:normalizedString">`
	`  <xs:enumeration value="p2p"/>`
	`  <xs:enumeration value="p2mp"/>`
	`  <xs:enumeration value="bcast"/>`
	`  </xs:restriction>`
	`</xs:simpleType>`

Simple Type `typeTxDemandPriority`

Namespace	DR-GW-Interface/DR-GW-Call.CommonTypes
Annotations	`Defines priority of speech item request: normal, pre-emptive, or emergency.`
Diagram	
Type	restriction of xs:normalizedString
Facets	enumeration `normal` enumeration `preemptive` enumeration `emergency`
Used by	Element typeCallAttributes/demandPriority

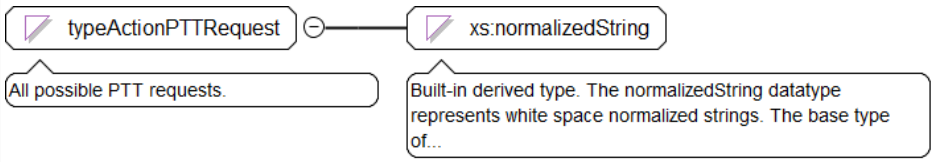
Source	`<xs:simpleType name="typeTxDemandPriority">`
	`  <xs:annotation>`
	`  <xs:documentation>Defines priority of speech item request: normal, pre-emptive, or emergency.</xs:documentation>`
	`  </xs:annotation>`
	`  <xs:restriction base="xs:normalizedString">`
	`  <xs:enumeration value="normal"/>`
	`  <xs:enumeration value="preemptive"/>`
	`  <xs:enumeration value="emergency"/>`
	`  </xs:restriction>`
	`</xs:simpleType>`

Simple Type `typeWorkstationId`

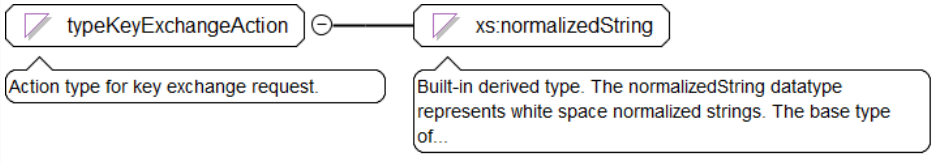
Namespace	DR-GW-Interface/DR-GW-Call.CommonTypes
Annotations	`Optional parameter is used to support the "neighbours" feature.`
Diagram	
Type	xs:normalizedString

Used by	Elements	Call_PTTRequest/workstationId, Call_Request/workstationId
Source	<xs:simpleType name="typeWorkstationId"> <xs:annotation> <xs:documentation>Optional parameter is used to support the "neighbours" feature.</xs:documentation> </xs:annotation> <xs:restriction base="xs:normalizedString"> </xs:restriction> </xs:simpleType>	

Simple Type `typeActionPTTRequest`

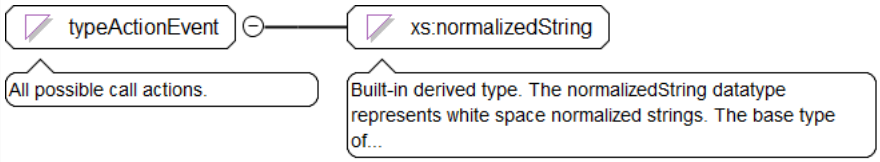
Namespace	DR-GW-Interface/DR-GW-Call.CommonTypes		
Annotations	All possible PTT requests.		
Diagram			
Type	restriction of xs:normalizedString		
Facets	enumeration	demandtx	This method can be used to request a speech item for a connected call.
	enumeration	ceasetx	This method is used to inform the system that the speech item is not needed any more.
Used by	Element	Call_PTTRequest/action	
Source	<xs:simpleType name="typeActionPTTRequest"> <xs:annotation> <xs:documentation>All possible PTT requests.</xs:documentation> </xs:annotation> <xs:restriction base="xs:normalizedString"> <xs:enumeration value="demandtx"> <xs:annotation> <xs:documentation>This method can be used to request a speech item for a connected call.</xs:documentation> </xs:annotation> </xs:enumeration> <xs:enumeration value="ceasetx"> <xs:annotation> <xs:documentation>This method is used to inform the system that the speech item is not needed any more.</xs:documentation> </xs:annotation> </xs:enumeration> </xs:restriction> </xs:simpleType>		

Simple Type `typeKeyExchangeAction`

Namespace	DR-GW-Interface/DR-GW-Call.CommonTypes		
Annotations	Action type for key exchange request.		
Diagram			
Type	restriction of xs:normalizedString		
Facets	enumeration	start	
	enumeration	stop	
Used by	Element	Call_KeyExchange/action	
Source	<xs:simpleType name="typeKeyExchangeAction"> <xs:annotation> <xs:documentation>Action type for key exchange request.</xs:documentation> </xs:annotation> <xs:restriction base="xs:normalizedString"> <xs:enumeration value="start"/> <xs:enumeration value="stop"/>		

```
  </xs:restriction>
</xs:simpleType>
```

Simple Type typeActionEvent

Namespace	DR-GW-Interface/DR-GW-Call.CommonTypes
Annotations	All possible call actions.
Diagram	
Type	restriction of xs:normalizedString

Facets

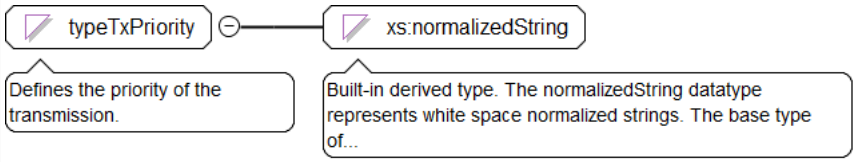
enumeration	incoming	This event fired when there is an incoming call. This is the first indication of a new incoming call.
enumeration	connected	This event is used to inform that call has been connected and call setup is finished.
enumeration	held	This event is used to inform TCS Client that individual call was put to hold.
enumeration	resumed	This event is used to inform that individual call has been taken from hold.
enumeration	disconnected	This event is used to inform that the call was disconnected.
enumeration	transferred	This event is a response to transfer method call and indicates the result of the request.

Source

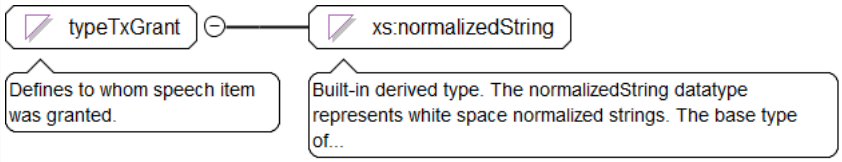
```
<xs:simpleType name="typeActionEvent">
  <xs:annotation>
    <xs:documentation>All possible call actions.</xs:documentation>
  </xs:annotation>
  <xs:restriction base="xs:normalizedString">
    <xs:enumeration value="incoming">
      <xs:annotation>
        <xs:documentation>This event fired when there is an incoming call. This is the first
 indication of a new incoming call.</xs:documentation>
      </xs:annotation>
    </xs:enumeration>
    <xs:enumeration value="connected">
      <xs:annotation>
        <xs:documentation>This event is used to inform that call has been connected and call setup
 is finished.</xs:documentation>
      </xs:annotation>
    </xs:enumeration>
    <xs:enumeration value="held">
      <xs:annotation>
        <xs:documentation>This event is used to inform TCS Client that individual call was put to
 hold.</xs:documentation>
      </xs:annotation>
    </xs:enumeration>
    <xs:enumeration value="resumed">
      <xs:annotation>
        <xs:documentation>This event is used to inform that individual call has been taken from
 hold.</xs:documentation>
      </xs:annotation>
    </xs:enumeration>
    <xs:enumeration value="disconnected">
      <xs:annotation>
        <xs:documentation>This event is used to inform that the call was disconnected.</
xs:documentation>
      </xs:annotation>
    </xs:enumeration>
    <xs:enumeration value="transferred">
      <xs:annotation>
        <xs:documentation>This event is a response to transfer method call and indicates the result
 of the request.</xs:documentation>
      </xs:annotation>
    </xs:enumeration>
  </xs:restriction>
</xs:simpleType>
```

Simple Type typeTxPriority

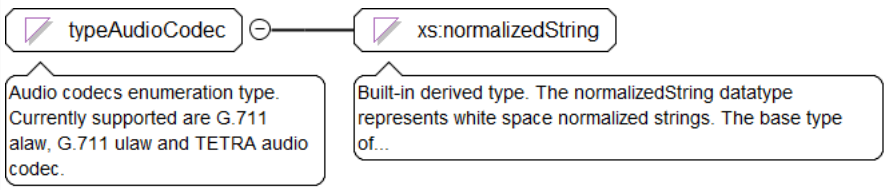
Namespace	DR-GW-Interface/DR-GW-Call.CommonTypes

Annotations	Defines the priority of the transmission.
Diagram	
Type	restriction of xs:normalizedString
Facets	enumeration normal enumeration emergency
Used by	Element typeTxGranted/txPriority
Source	<xs:simpleType name="typeTxPriority"> <xs:annotation> <xs:documentation>Defines the priority of the transmission.</xs:documentation> </xs:annotation> <xs:restriction base="xs:normalizedString"> <xs:enumeration value="normal"/> <xs:enumeration value="emergency"/> </xs:restriction> </xs:simpleType>

Simple Type typeTxGrant

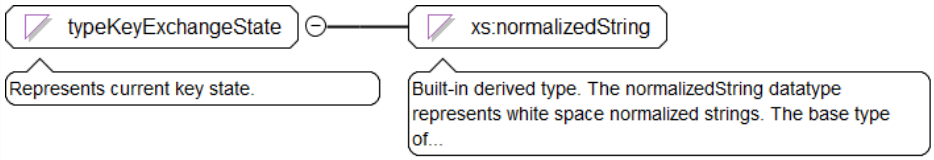
Namespace	DR-GW-Interface/DR-GW-Call.CommonTypes
Annotations	Defines to whom speech item was granted.
Diagram	
Type	restriction of xs:normalizedString
Facets	enumeration granted enumeration notGranted enumeration queued enumeration granted2another
Used by	Element typeTxGranted/txGrant
Source	<xs:simpleType name="typeTxGrant"> <xs:annotation> <xs:documentation>Defines to whom speech item was granted.</xs:documentation> </xs:annotation> <xs:restriction base="xs:normalizedString"> <xs:enumeration value="granted"/> <xs:enumeration value="notGranted"/> <xs:enumeration value="queued"/> <xs:enumeration value="granted2another"/> </xs:restriction> </xs:simpleType>

Simple Type typeAudioCodec

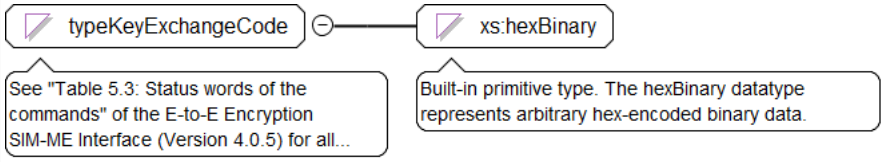
Namespace	DR-GW-Interface/DR-GW-Call.CommonTypes
Annotations	Audio codecs enumeration type. Currently supported are G.711 alaw, G.711 ulaw and TETRA audio codec.
Diagram	
Type	restriction of xs:normalizedString

Facets	enumeration	alaw
	enumeration	ulaw
	enumeration	tetra
Source	<xs:simpleType name="typeAudioCodec"> <xs:annotation> <xs:documentation>Audio codecs enumeration type. Currently supported are G.711 alaw, G.711 ulaw and TETRA audio codec.</xs:documentation> </xs:annotation> <xs:restriction base="xs:normalizedString"> <xs:enumeration value="alaw"/> <xs:enumeration value="ulaw"/> <xs:enumeration value="tetra"/> </xs:restriction> </xs:simpleType>	

Simple Type typeKeyExchangeState

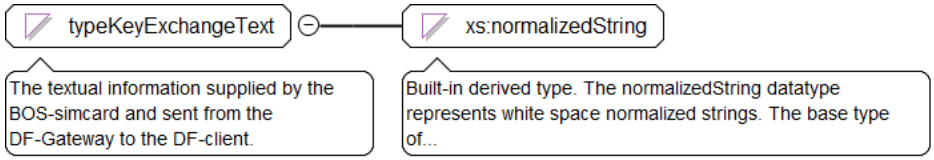
Namespace	DR-GW-Interface/DR-GW-Call.CommonTypes		
Annotations	Represents current key state.		
Diagram			
Type	restriction of xs:normalizedString		
Facets	enumeration	keyValid	current key is valid, no user action required.
	enumeration	keyInvalid	Key invalid, user must request key exchange.
	enumeration	keyExchangeInProgress	Key exchange in progress, user may abort exchange or wait until it gets finished.
Source	<xs:simpleType name="typeKeyExchangeState"> <xs:annotation> <xs:documentation>Represents current key state.</xs:documentation> </xs:annotation> <xs:restriction base="xs:normalizedString"> <xs:enumeration value="keyValid"> <xs:annotation> <xs:documentation>current key is valid, no user action required.</xs:documentation> </xs:annotation> </xs:enumeration> <xs:enumeration value="keyInvalid"> <xs:annotation> <xs:documentation>Key invalid, user must request key exchange.</xs:documentation> </xs:annotation> </xs:enumeration> <xs:enumeration value="keyExchangeInProgress"> <xs:annotation> <xs:documentation>Key exchange in progress, user may abort exchange or wait until it gets finished.</xs:documentation> </xs:annotation> </xs:enumeration> </xs:restriction> </xs:simpleType>		

Simple Type typeKeyExchangeCode

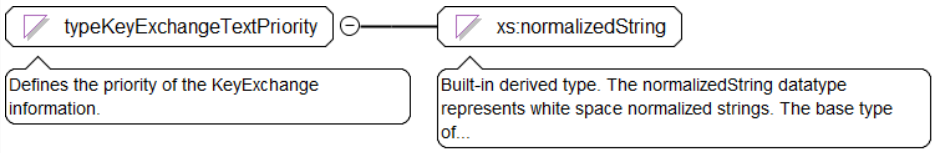
Namespace	DR-GW-Interface/DR-GW-Call.CommonTypes
Annotations	See "Table 5.3: Status words of the commands" of the E-to-E Encryption SIM-ME Interface (Version 4.0.5) for all possible code values.
Diagram	
Type	restriction of xs:hexBinary

Facets	length 2
Source	<xs:simpleType name="typeKeyExchangeCode"> <xs:annotation> <xs:documentation>See "Table 5.3: Status words of the commands" of the E-to-E Encryption SIM-ME Interface (Version 4.0.5) for all possible code values.</xs:documentation> </xs:annotation> <xs:restriction base="xs:hexBinary"> <xs:length value="2"/> </xs:restriction> </xs:simpleType>

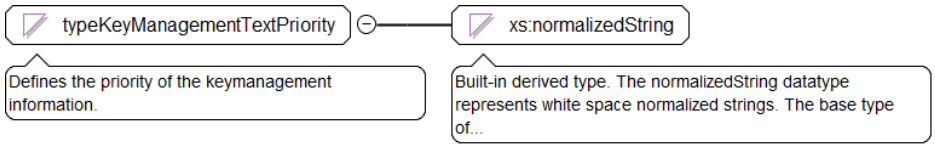
Simple Type typeKeyExchangeText

Namespace	DR-GW-Interface/DR-GW-Call.CommonTypes
Annotations	The textual information supplied by the BOS-simcard and sent from the DF-Gateway to the DF-client.
Diagram	
Type	restriction of xs:normalizedString
Facets	maxLength 100
Source	<xs:simpleType name="typeKeyExchangeText"> <xs:annotation> <xs:documentation>The textual information supplied by the BOS-simcard and sent from the DF-Gateway to the DF-client.</xs:documentation> </xs:annotation> <xs:restriction base="xs:normalizedString"> <xs:maxLength value="100"/> </xs:restriction> </xs:simpleType>

Simple Type typeKeyExchangeTextPriority

Namespace	DR-GW-Interface/DR-GW-Call.CommonTypes
Annotations	Defines the priority of the KeyExchange information.
Diagram	
Type	restriction of xs:normalizedString
Facets	enumeration normal enumeration high
Source	<xs:simpleType name="typeKeyExchangeTextPriority"> <xs:annotation> <xs:documentation>Defines the priority of the KeyExchange information.</xs:documentation> </xs:annotation> <xs:restriction base="xs:normalizedString"> <xs:enumeration value="normal"/> <xs:enumeration value="high"/> </xs:restriction> </xs:simpleType>

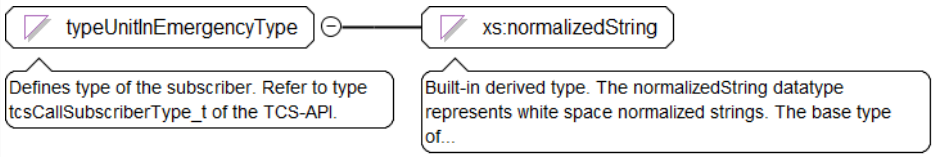
Simple Type typeKeyManagementTextPriority

Namespace	DR-GW-Interface/DR-GW-Call.CommonTypes
Annotations	Defines the priority of the keymanagement information.
Diagram	

Type	restriction of xs:normalizedString
Facets	enumeration `normal` enumeration `high`

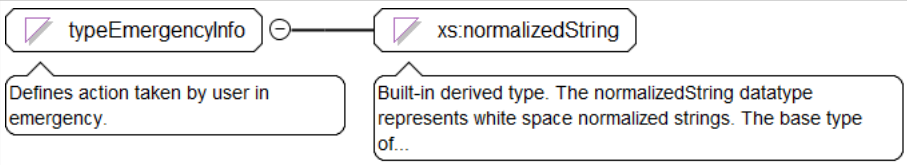
Source	`<xs:simpleType name="typeKeyManagementTextPriority">`
	`  <xs:annotation>`
	`  <xs:documentation>Defines the priority of the keymanagement information.</xs:documentation>`
	`  </xs:annotation>`
	`  <xs:restriction base="xs:normalizedString">`
	`  <xs:enumeration value="normal"/>`
	`  <xs:enumeration value="high"/>`
	`  </xs:restriction>`
	`</xs:simpleType>`

Simple Type `typeUnitInEmergencyType`

Namespace	DR-GW-Interface/DR-GW-Call.CommonTypes
Annotations	Defines type of the subscriber. Refer to type tcsCallSubscriberType_t of the TCS-API.
Diagram	
Type	restriction of xs:normalizedString
Facets	enumeration `dummy` enumeration `ms` enumeration `g4wif` enumeration `external` enumeration `ws`

Source	`<xs:simpleType name="typeUnitInEmergencyType">`
	`  <xs:annotation>`
	`  <xs:documentation>Defines type of the subscriber. Refer to type tcsCallSubscriberType_t of the TCS-API.</xs:documentation>`
	`  </xs:annotation>`
	`  <xs:restriction base="xs:normalizedString">`
	`  <xs:enumeration value="dummy"/>`
	`  <xs:enumeration value="ms"/>`
	`  <xs:enumeration value="g4wif"/>`
	`  <xs:enumeration value="external"/>`
	`  <xs:enumeration value="ws"/>`
	`  </xs:restriction>`
	`</xs:simpleType>`

Simple Type `typeEmergencyInfo`

Namespace	DR-GW-Interface/DR-GW-Call.CommonTypes
Annotations	Defines action taken by user in emergency.
Diagram	
Type	restriction of xs:normalizedString
Facets	enumeration `addTx` enumeration `add` enumeration `ceased` enumeration `demandTx` enumeration `removed` enumeration `emergencyCallDisconnected`
Source	`<xs:simpleType name="typeEmergencyInfo">`

```
  <xs:annotation>
    <xs:documentation>Defines action taken by user in emergency.</xs:documentation>
  </xs:annotation>
  <xs:restriction base="xs:normalizedString">
    <xs:enumeration value="addTx"/>
    <xs:enumeration value="add"/>
    <xs:enumeration value="ceased"/>
    <xs:enumeration value="demandTx"/>
    <xs:enumeration value="removed"/>
    <xs:enumeration value="emergencyCallDisconnected"/>
  </xs:restriction>
</xs:simpleType>
```