Schema

SUTI MessageSchema

schema location: L:\Admin\SUTI\Standard\SUTI V 2.0.0\SUTI_MessageXSD_2_0_0.xsd
attribute form default: **unqualified**
element form default: **qualified**

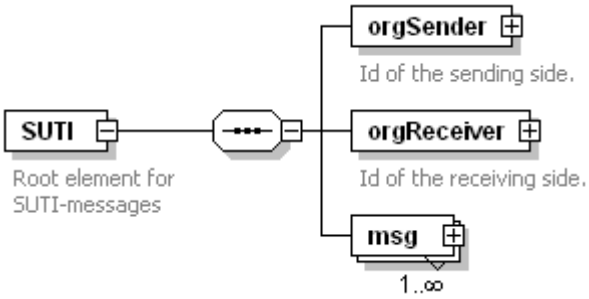
Elements
SUTI

Complex types
addressType
agreement
associatedReservation
attribute
attributesType
cancellationConsequence
capacity
connection
contactInfo
contactInfosType
content
contents
driver
economyType
exchangeRates
exhangeRate
formOfPayment
geographicLocation
idEkInfo
idMsgRef
idType
manualDescriptionType
msg
multiDispatch
node
order
orderReject
orgType
payment
pickupConfirmation
position
price
priceCalculation
process
product
referencesTo
resourceType
route
seats
subOrderType
taxiMeter
time
timesType

vehicle

element **SUTI**

diagram



properties content complex

children **orgSender** **orgReceiver** **msg**

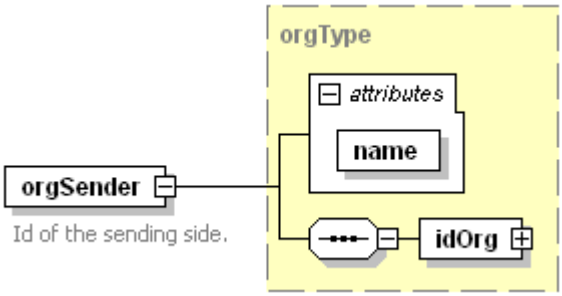
annotation documentation
Root element for SUTI-messages

source

```
<xs:element name="SUTI">
 <xs:annotation>
  <xs:documentation>Root element for SUTI-messages</xs:documentation>
 </xs:annotation>
 <xs:complexType>
  <xs:sequence>
   <xs:element name="orgSender" type="orgType">
    <xs:annotation>
     <xs:documentation>Id of the sending side. </xs:documentation>
    </xs:annotation>
   </xs:element>
   <xs:element name="orgReceiver" type="orgType">
    <xs:annotation>
     <xs:documentation>Id of the receiving side.</xs:documentation>
    </xs:annotation>
   </xs:element>
   <xs:element name="msg" maxOccurs="unbounded">
    <xs:complexType>
     <xs:complexContent>
      <xs:extension base="msg"/>
     </xs:complexContent>
    </xs:complexType>
   </xs:element>
  </xs:sequence>
 </xs:complexType>
</xs:element>
```

element **SUTI/orgSender**

diagram



type **orgType**

properties
isRef 0
content complex

children **idOrg**

attributes

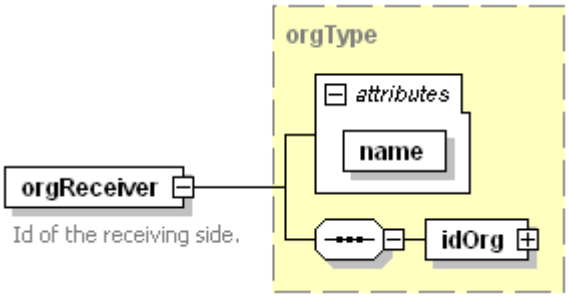
Name	Type	Use	Default	Fixed	annotation
name	**xs:string**	required			

annotation
documentation
Id of the sending side.

source
```
<xs:element name="orgSender" type="orgType">
 <xs:annotation>
  <xs:documentation>Id of the sending side. </xs:documentation>
 </xs:annotation>
</xs:element>
```

element **SUTI/orgReceiver**

diagram



type **orgType**

properties
isRef 0
content complex

children **idOrg**

attributes

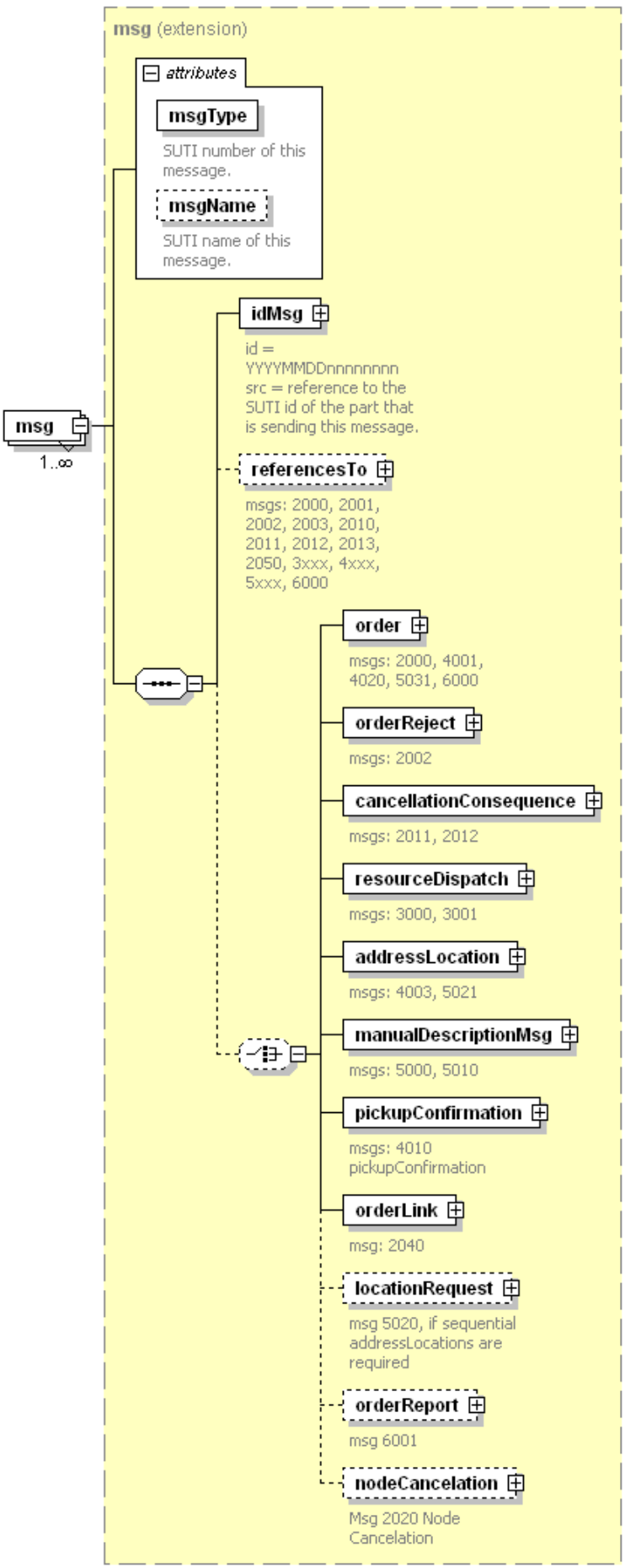
Name	Type	Use	Default	Fixed	annotation
name	**xs:string**	required			

annotation
documentation
Id of the receiving side.

source
```
<xs:element name="orgReceiver" type="orgType">
 <xs:annotation>
  <xs:documentation>Id of the receiving side.</xs:documentation>
 </xs:annotation>
</xs:element>
```

element **SUTI/msg**

diagram


msg (extension)
attributes
msgType
SUTI number of this message.
msgName
SUTI name of this message.
idMsg
id = YYYYMMDDnnnnnnnn
src = reference to the SUTI id of the part that is sending this message.
msg
1..∞
referencesTo
msgs: 2000, 2001, 2002, 2003, 2010, 2011, 2012, 2013, 2050, 3xxx, 4xxx, 5xxx, 6000
order
msgs: 2000, 4001, 4020, 5031, 6000
orderReject
msgs: 2002
cancellationConsequence
msgs: 2011, 2012
resourceDispatch
msgs: 3000, 3001
addressLocation
msgs: 4003, 5021
manualDescriptionMsg
msgs: 5000, 5010
pickupConfirmation
msgs: 4010 pickupConfirmation
orderLink
msg: 2040
locationRequest
msg 5020, if sequential addressLocations are required
orderReport
msg 6001
nodeCancelation
Msg 2020 Node Cancelation

type extension of **msg**

properties
- isRef 0
- minOcc 1
- maxOcc unbounded
- content complex

children **idMsg** **referencesTo** **order** **orderReject** **cancellationConsequence** **resourceDispatch** **addressLocation** **manualDescriptionMsg** **pickupConfirmation** **orderLink** **locationRequest** **orderReport** **nodeCancelation**

attributes

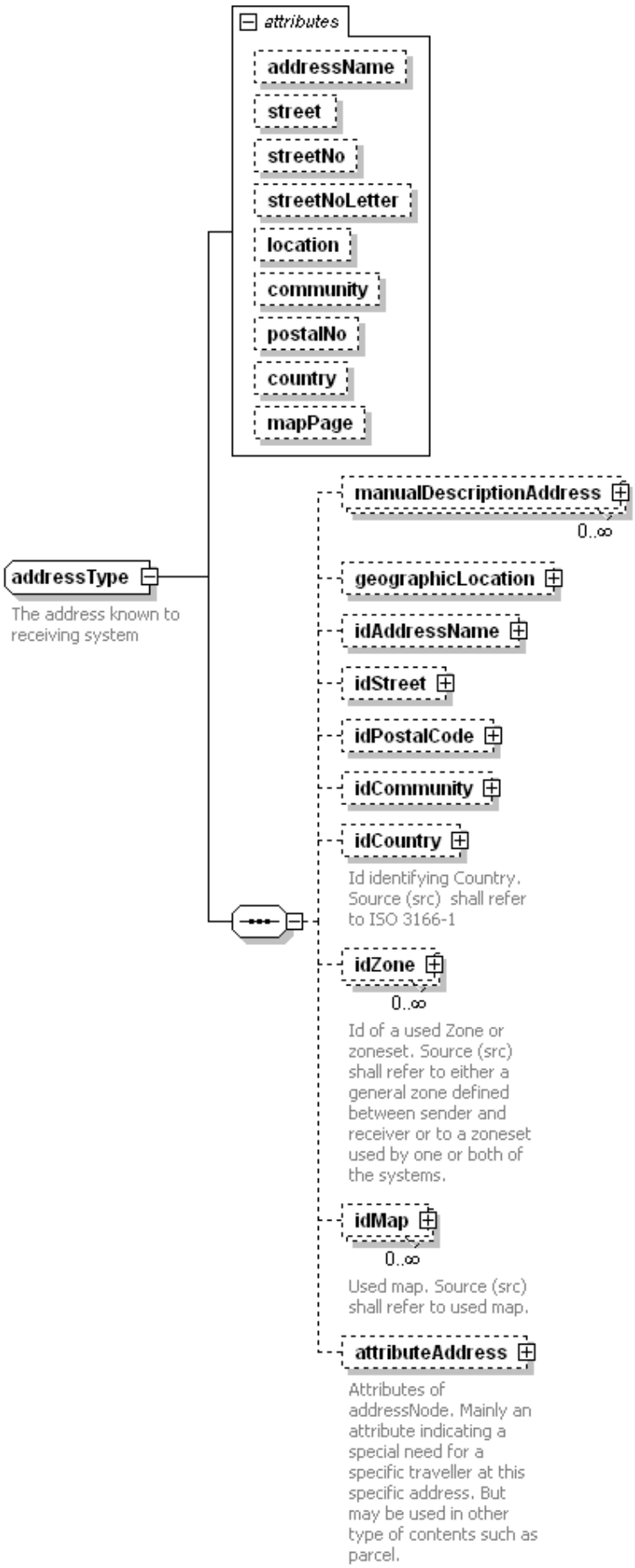
Name	Type	Use	Default	Fixed	annotation
msgType	**xs:string**	required			documentation SUTI number of this message.
msgName	**xs:string**	optional			documentation SUTI name of this message.

source

```
<xs:element name="msg" maxOccurs="unbounded">
  <xs:complexType>
    <xs:complexContent>
      <xs:extension base="msg"/>
    </xs:complexContent>
  </xs:complexType>
</xs:element>
```

complexType **addressType**

diagram


attributes
addressName
street
streetNo
streetNoLetter
location
community
postalNo
country
mapPage
addressType
The address known to receiving system
manualDescriptionAddress
0..∞
geographicLocation
idAddressName
idStreet
idPostalCode
idCommunity
idCountry
Id identifying Country. Source (src) shall refer to ISO 3166-1
idZone
0..∞
Id of a used Zone or zoneset. Source (src) shall refer to either a general zone defined between sender and receiver or to a zoneset used by one or both of the systems.
idMap
0..∞
Used map. Source (src) shall refer to used map.
attributeAddress
Attributes of addressNode. Mainly an attribute indicating a special need for a specific traveller at this specific address. But may be used in other type of contents such as parcel.

children **manualDescriptionAddress geographicLocation idAddressName idStreet idPostalCode idCommunity idCountry idZone idMap attributeAddress**

used by elements **msg/addressLocation node/addressNode associatedReservation/addressReservation**

attributes

Name	Type	Use	Default	Fixed	annotation
addressName	**xs:string**	optional			
street	**xs:string**	optional			
streetNo	**xs:positiveInteger**	optional			
streetNoLetter	**xs:string**	optional			
location	**xs:string**	optional			
community	**xs:string**	optional			
postalNo	**xs:string**	optional			
country	**xs:string**	optional			
mapPage	**xs:string**	optional			

annotation documentation

The address known to receiving system

source

```
<xs:complexType name="addressType">
 <xs:annotation>
  <xs:documentation>The address known to receiving system</xs:documentation>
 </xs:annotation>
 <xs:sequence>
  <xs:element name="manualDescriptionAddress" type="manualDescriptionType" minOccurs="0"
maxOccurs="unbounded"/>
  <xs:element name="geographicLocation" type="geographicLocation" minOccurs="0"/>
  <xs:element name="idAddressName" type="idType" minOccurs="0"/>
  <xs:element name="idStreet" minOccurs="0">
   <xs:complexType>
    <xs:complexContent>
     <xs:extension base="idType"/>
    </xs:complexContent>
   </xs:complexType>
  </xs:element>
  <xs:element name="idPostalCode" type="idType" minOccurs="0"/>
  <xs:element name="idCommunity" type="idType" minOccurs="0"/>
  <xs:element name="idCountry" type="idType" minOccurs="0">
   <xs:annotation>
    <xs:documentation>Id identifying Country. Source (src)  shall refer to ISO 3166-1</xs:documentation>
   </xs:annotation>
  </xs:element>
  <xs:element name="idZone" type="idType" minOccurs="0" maxOccurs="unbounded">
   <xs:annotation>
    <xs:documentation>Id of a used Zone or zoneset. Source (src)  shall refer to either a general zone
defined between sender and receiver or to a zoneset used by one or both of the
systems.</xs:documentation>
   </xs:annotation>
  </xs:element>
  <xs:element name="idMap" type="idType" minOccurs="0" maxOccurs="unbounded">
   <xs:annotation>
    <xs:documentation>Used map. Source (src)  shall refer to used map.</xs:documentation>
   </xs:annotation>
  </xs:element>
  <xs:element name="attributeAddress" type="attributesType" minOccurs="0">
   <xs:annotation>
    <xs:documentation>Attributes of addressNode. Mainly an attribute indicating a special need for a
specific traveller at this specific address. But may be used in other type of contents such as
parcel.</xs:documentation>
   </xs:annotation>
  </xs:element>
 </xs:sequence>
 <xs:attribute name="addressName" type="xs:string" use="optional"/>
 <xs:attribute name="street" type="xs:string" use="optional"/>
 <xs:attribute name="streetNo" type="xs:positiveInteger" use="optional"/>
 <xs:attribute name="streetNoLetter" type="xs:string" use="optional"/>
 <xs:attribute name="location" type="xs:string" use="optional"/>
 <xs:attribute name="community" type="xs:string" use="optional"/>
```

```
    <xs:attribute name="postalNo" type="xs:string" use="optional"/>
    <xs:attribute name="country" type="xs:string" use="optional"/>
    <xs:attribute name="mapPage" type="xs:string" use="optional"/>
  </xs:complexType>
```

attribute **addressType/@addressName**

type **xs:string**

properties
isRef 0
use optional

source
```
<xs:attribute name="addressName" type="xs:string" use="optional"/>
```

attribute **addressType/@street**

type **xs:string**

properties
isRef 0
use optional

source
```
<xs:attribute name="street" type="xs:string" use="optional"/>
```

attribute **addressType/@streetNo**

type **xs:positiveInteger**

properties
isRef 0
use optional

source
```
<xs:attribute name="streetNo" type="xs:positiveInteger" use="optional"/>
```

attribute **addressType/@streetNoLetter**

type **xs:string**

properties
isRef 0
use optional

source
```
<xs:attribute name="streetNoLetter" type="xs:string" use="optional"/>
```

attribute **addressType/@location**

type **xs:string**

properties
isRef 0
use optional

source
```
<xs:attribute name="location" type="xs:string" use="optional"/>
```

attribute **addressType/@community**

type **xs:string**

properties
isRef 0
use optional

source
```
<xs:attribute name="community" type="xs:string" use="optional"/>
```

attribute **addressType/@postalNo**

type **xs:string**

properties
isRef 0
use optional

source <xs:attribute name="postalNo" type="xs:string" use="optional"/>

attribute **addressType/@country**

type **xs:string**

properties isRef 0
use optional

source <xs:attribute name="country" type="xs:string" use="optional"/>

attribute **addressType/@mapPage**

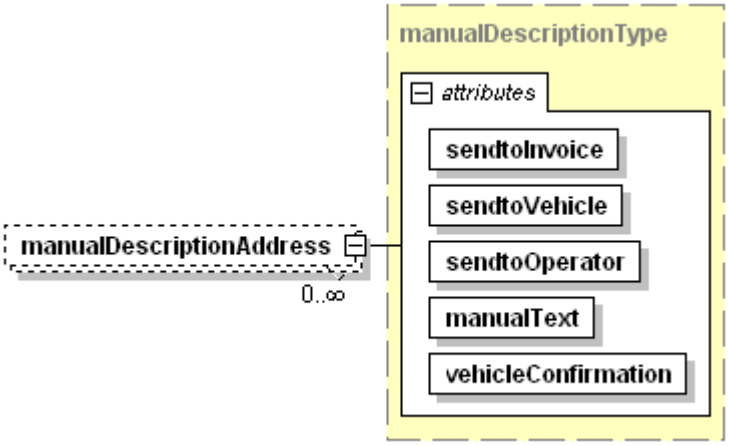
type **xs:string**

properties isRef 0
use optional

source <xs:attribute name="mapPage" type="xs:string" use="optional"/>

element **addressType/manualDescriptionAddress**

diagram



type **manualDescriptionType**

properties isRef 0
minOcc 0
maxOcc unbounded
content complex

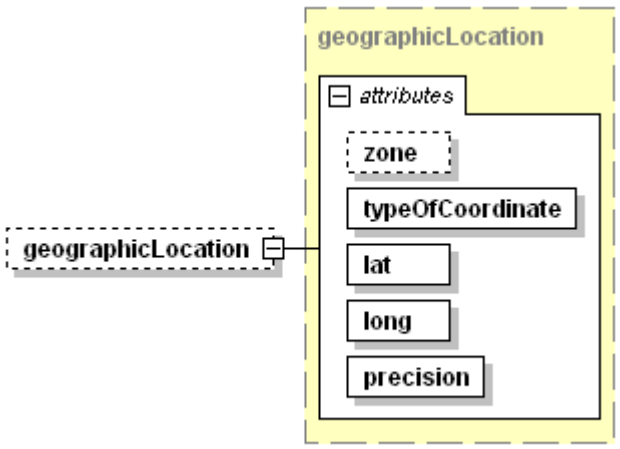
attributes

Name	Type	Use	Default	Fixed	annotation
sendtoInvoice	**xs:boolean**	required			
sendtoVehicle	**xs:boolean**	required			
sendtoOperator	**xs:boolean**	required			
manualText	**xs:string**	required			
vehicleConfirmation	**xs:boolean**	required			

source <xs:element name="manualDescriptionAddress" type="manualDescriptionType" minOccurs="0" maxOccurs="unbounded"/>

element **addressType/geographicLocation**

diagram



type **geographicLocation**

properties
- isRef 0
- minOcc 0
- maxOcc 1
- content complex

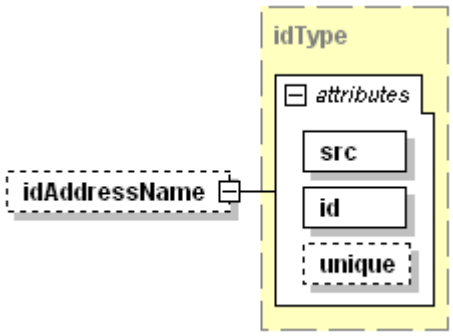
attributes

Name	Type	Use	Default	Fixed	annotation
zone	**xs:string**	optional			
typeOfCoordinate	**xs:string**	required			
lat	**xs:float**	required			
long	**xs:float**	required			
precision	**xs:integer**	required			

source
```
<xs:element name="geographicLocation" type="geographicLocation" minOccurs="0"/>
```

element **addressType/idAddressName**

diagram



type **idType**

properties
- isRef 0
- minOcc 0
- maxOcc 1
- content complex

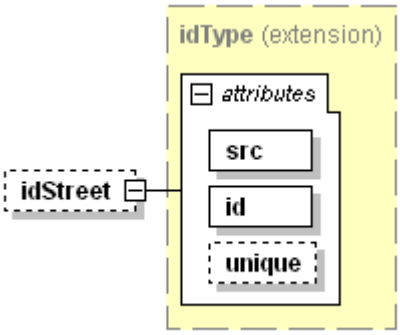
attributes

Name	Type	Use	Default	Fixed	annotation
src	**xs:string**	required			
id	**xs:string**	required			
unique	**xs:boolean**	optional	false		

source
```
<xs:element name="idAddressName" type="idType" minOccurs="0"/>
```

element **addressType/idStreet**

diagram



type extension of **idType**

properties

isRef	0
minOcc	0
maxOcc	1
content	complex

attributes

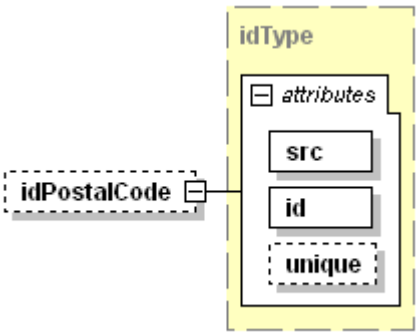
Name	Type	Use	Default	Fixed	annotation
src	**xs:string**	required			
id	**xs:string**	required			
unique	**xs:boolean**	optional	false		

source

```
<xs:element name="idStreet" minOccurs="0">
 <xs:complexType>
  <xs:complexContent>
   <xs:extension base="idType"/>
  </xs:complexContent>
 </xs:complexType>
</xs:element>
```

element **addressType/idPostalCode**

diagram



type **idType**

properties

isRef	0
minOcc	0
maxOcc	1
content	complex

attributes

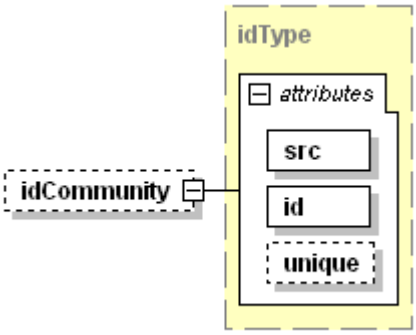
Name	Type	Use	Default	Fixed	annotation
src	**xs:string**	required			
id	**xs:string**	required			
unique	**xs:boolean**	optional	false		

source

```
<xs:element name="idPostalCode" type="idType" minOccurs="0"/>
```

element **addressType/idCommunity**

diagram



type **idType**

properties
- isRef 0
- minOcc 0
- maxOcc 1
- content complex

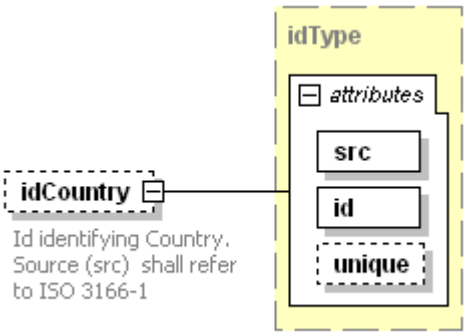
attributes

Name	Type	Use	Default	Fixed	annotation
src	**xs:string**	required			
id	**xs:string**	required			
unique	**xs:boolean**	optional	false		

source
```
<xs:element name="idCommunity" type="idType" minOccurs="0"/>
```

element **addressType/idCountry**

diagram



type **idType**

properties
- isRef 0
- minOcc 0
- maxOcc 1
- content complex

attributes

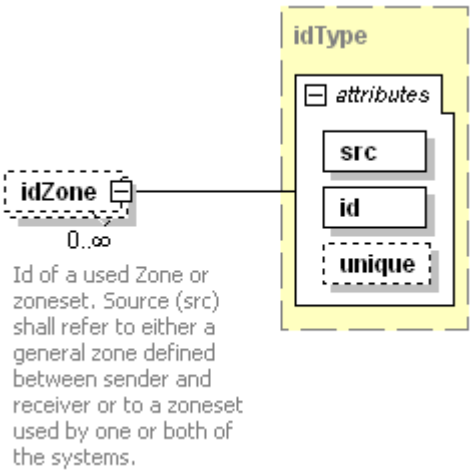
Name	Type	Use	Default	Fixed	annotation
src	**xs:string**	required			
id	**xs:string**	required			
unique	**xs:boolean**	optional	false		

annotation
documentation
Id identifying Country. Source (src) shall refer to ISO 3166-1

source
```
<xs:element name="idCountry" type="idType" minOccurs="0">
 <xs:annotation>
  <xs:documentation>Id identifying Country. Source (src)  shall refer to ISO 3166-1</xs:documentation>
 </xs:annotation>
</xs:element>
```

element **addressType/idZone**

diagram



type **idType**

properties

isRef	0
minOcc	0
maxOcc	unbounded
content	complex

attributes

Name	Type	Use	Default	Fixed	annotation
src	**xs:string**	required			
id	**xs:string**	required			
unique	**xs:boolean**	optional	false		

annotation

documentation

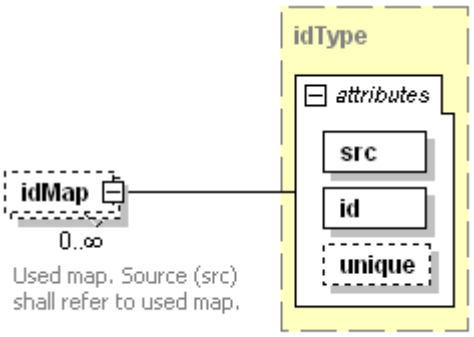
Id of a used Zone or zoneset. Source (src) shall refer to either a general zone defined between sender and receiver or to a zoneset used by one or both of the systems.

source

```
<xs:element name="idZone" type="idType" minOccurs="0" maxOccurs="unbounded">
 <xs:annotation>
  <xs:documentation>Id of a used Zone or zoneset. Source (src) shall refer to either a general zone defined between sender and receiver or to a zoneset used by one or both of the systems.</xs:documentation>
 </xs:annotation>
</xs:element>
```

element **addressType/idMap**

diagram



type **idType**

properties

isRef	0
minOcc	0
maxOcc	unbounded
content	complex

attributes

Name	Type	Use	Default	Fixed	annotation

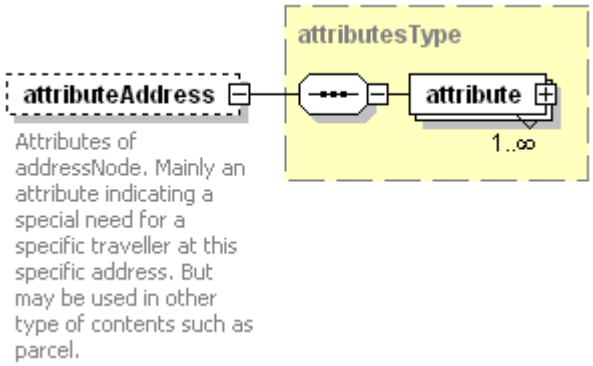
	src	**xs:string**	required	
	id	**xs:string**	required	
	unique	**xs:boolean**	optional	false

annotation
documentation
Used map. Source (src) shall refer to used map.

source
```
<xs:element name="idMap" type="idType" minOccurs="0" maxOccurs="unbounded">
 <xs:annotation>
  <xs:documentation>Used map. Source (src) shall refer to used map.</xs:documentation>
 </xs:annotation>
</xs:element>
```

element **addressType/attributeAddress**

diagram



type **attributesType**

properties
isRef 0
minOcc 0
maxOcc 1
content complex

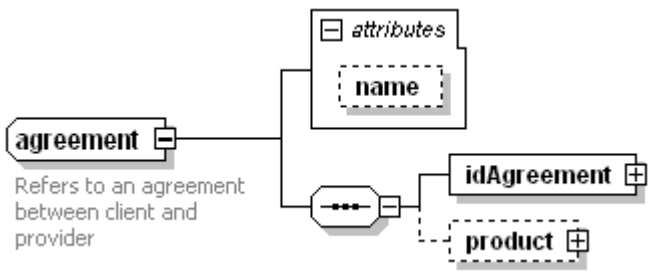
children **attribute**

annotation
documentation
Attributes of addressNode. Mainly an attribute indicating a special need for a specific traveller at this specific address. But may be used in other type of contents such as parcel.

source
```
<xs:element name="attributeAddress" type="attributesType" minOccurs="0">
 <xs:annotation>
  <xs:documentation>Attributes of addressNode. Mainly an attribute indicating a special need for a specific traveller at this specific address. But may be used in other type of contents such as parcel.</xs:documentation>
 </xs:annotation>
</xs:element>
```

complexType **agreement**

diagram



children **idAgreement** **product**

used by
element **order/agreement**

attributes

Name	Type	Use	Default	Fixed	annotation
name	**xs:string**	optional			

annotation documentation
Refers to an agreement between client and provider

source
```
<xs:complexType name="agreement">
 <xs:annotation>
  <xs:documentation>Refers to an agreement between client and provider</xs:documentation>
 </xs:annotation>
 <xs:sequence>
  <xs:element name="idAgreement" type="idType"/>
  <xs:element name="product" type="product" minOccurs="0"/>
 </xs:sequence>
 <xs:attribute name="name" type="xs:string" use="optional"/>
</xs:complexType>
```

attribute **agreement/@name**

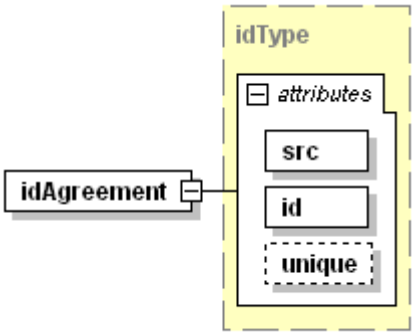
type **xs:string**

properties
isRef 0
use optional

source
```
<xs:attribute name="name" type="xs:string" use="optional"/>
```

element **agreement/idAgreement**

diagram



type **idType**

properties
isRef 0
content complex

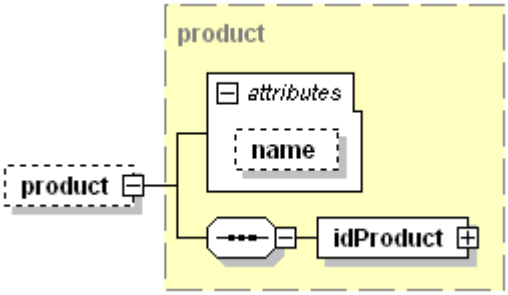
attributes

Name	Type	Use	Default	Fixed	annotation
src	**xs:string**	required			
id	**xs:string**	required			
unique	**xs:boolean**	optional	false		

source
```
<xs:element name="idAgreement" type="idType"/>
```

element **agreement/product**

diagram



type **product**

properties
isRef 0
minOcc 0
maxOcc 1
content complex

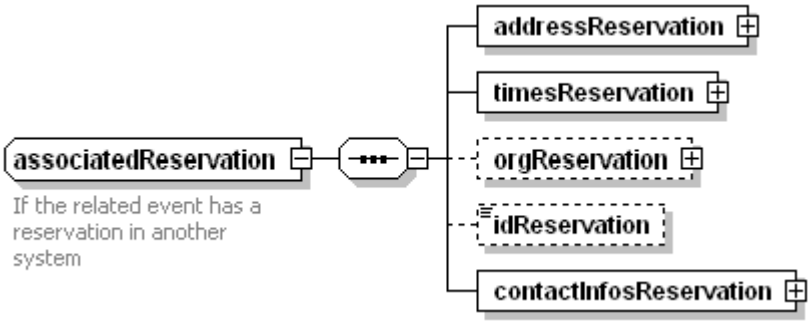
children **idProduct**

attributes	Name	Type	Use	Default	Fixed	annotation
	name	**xs:string**	optional			

source
```
<xs:element name="product" type="product" minOccurs="0"/>
```

complexType **associatedReservation**

diagram



children **addressReservation** **timesReservation** **orgReservation** **idReservation** **contactInfosReservation**

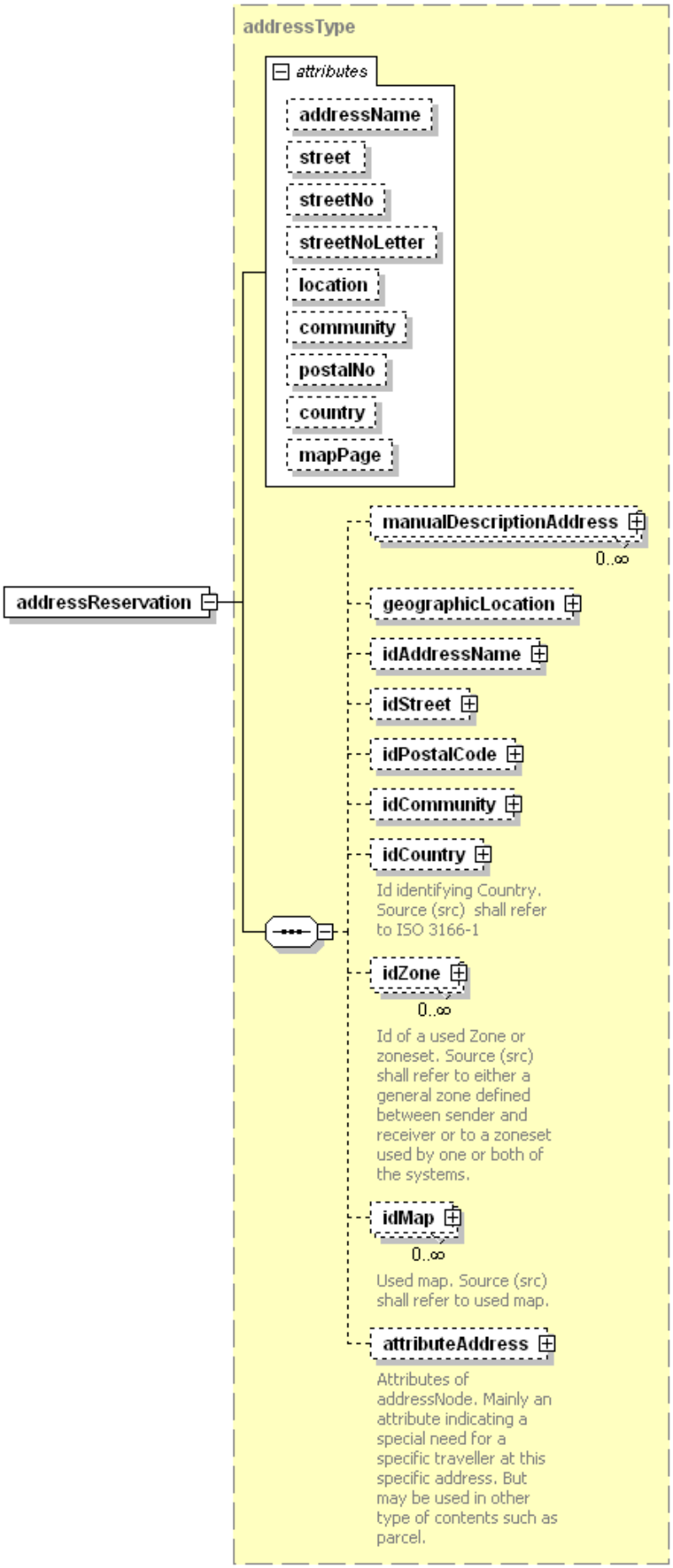
used by element **connection/associatedReservation**

annotation
documentation
If the related event has a reservation in another system

source
```
<xs:complexType name="associatedReservation">
 <xs:annotation>
  <xs:documentation>If the related event has a reservation in another system</xs:documentation>
 </xs:annotation>
 <xs:sequence>
  <xs:element name="addressReservation" type="addressType"/>
  <xs:element name="timesReservation" type="timesType"/>
  <xs:element name="orgReservation" type="orgType" minOccurs="0"/>
  <xs:element name="idReservation" type="xs:string" minOccurs="0"/>
  <xs:element name="contactInfosReservation" type="contactInfosType"/>
 </xs:sequence>
</xs:complexType>
```

element **associatedReservation/addressReservation**

diagram


addressType
attributes
addressName
street
streetNo
streetNoLetter
location
community
postalNo
country
mapPage
addressReservation
manualDescriptionAddress
0..∞
geographicLocation
idAddressName
idStreet
idPostalCode
idCommunity
idCountry
Id identifying Country. Source (src) shall refer to ISO 3166-1
idZone
0..∞
Id of a used Zone or zoneset. Source (src) shall refer to either a general zone defined between sender and receiver or to a zoneset used by one or both of the systems.
idMap
0..∞
Used map. Source (src) shall refer to used map.
attributeAddress
Attributes of addressNode. Mainly an attribute indicating a special need for a specific traveller at this specific address. But may be used in other type of contents such as parcel.

type **addressType**

properties
isRef 0
content complex

children **manualDescriptionAddress** **geographicLocation** **idAddressName** **idStreet** **idPostalCode** **idCommunity** **idCountry** **idZone** **idMap** **attributeAddress**

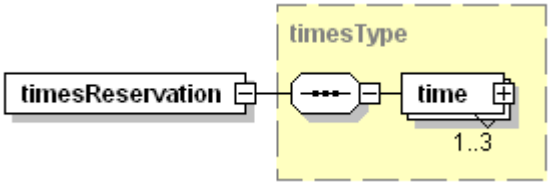
attributes

Name	Type	Use	Default	Fixed	annotation
addressName	**xs:string**	optional			
street	**xs:string**	optional			
streetNo	**xs:positiveInteger**	optional			
streetNoLetter	**xs:string**	optional			
location	**xs:string**	optional			
community	**xs:string**	optional			
postalNo	**xs:string**	optional			
country	**xs:string**	optional			
mapPage	**xs:string**	optional			

source
```
<xs:element name="addressReservation" type="addressType"/>
```

element **associatedReservation/timesReservation**

diagram



type **timesType**

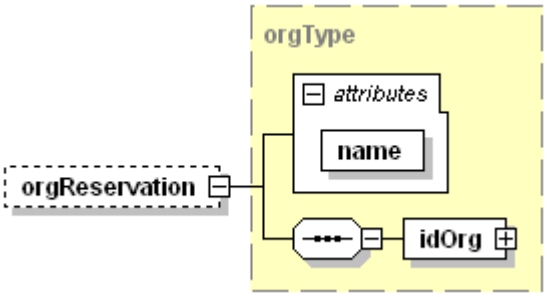
properties
isRef 0
content complex

children **time**

source
```
<xs:element name="timesReservation" type="timesType"/>
```

element **associatedReservation/orgReservation**

diagram



type **orgType**

properties
isRef 0
minOcc 0
maxOcc 1
content complex

children **idOrg**

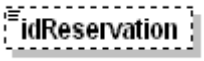
attributes

Name	Type	Use	Default	Fixed	annotation
name	**xs:string**	required			

source
```
<xs:element name="orgReservation" type="orgType" minOccurs="0"/>
```

element **associatedReservation/idReservation**

diagram



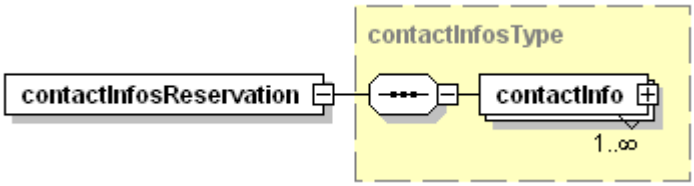
type **xs:string**

properties
- isRef 0
- minOcc 0
- maxOcc 1
- content simple

source
```
<xs:element name="idReservation" type="xs:string" minOccurs="0"/>
```

element **associatedReservation/contactInfosReservation**

diagram



type **contactInfosType**

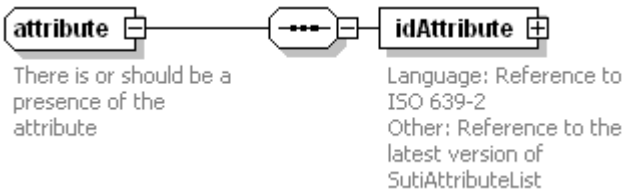
properties
- isRef 0
- content complex

children **contactInfo**

source
```
<xs:element name="contactInfosReservation" type="contactInfosType"/>
```

complexType **attribute**

diagram



children **idAttribute**

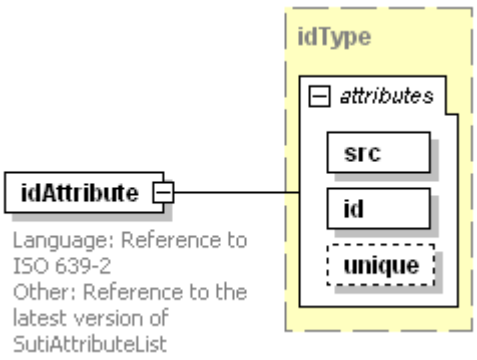
used by
- element **attributesType/attribute**

annotation
documentation
There is or should be a presence of the attribute

source
```
<xs:complexType name="attribute">
 <xs:annotation>
  <xs:documentation>There is or should be a presence of the attribute</xs:documentation>
 </xs:annotation>
 <xs:sequence>
  <xs:element name="idAttribute" type="idType">
   <xs:annotation>
    <xs:documentation>Language: Reference to ISO 639-2
Other: Reference to the latest version of SutiAttributeList</xs:documentation>
   </xs:annotation>
  </xs:element>
 </xs:sequence>
</xs:complexType>
```

element **attribute/idAttribute**

diagram



type **idType**

properties
- isRef 0
- content complex

attributes

Name	Type	Use	Default	Fixed	annotation
src	**xs:string**	required			
id	**xs:string**	required			
unique	**xs:boolean**	optional	false		

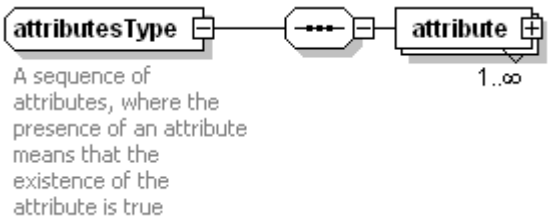
annotation
documentation
Language: Reference to ISO 639-2
Other: Reference to the latest version of SutiAttributeList

source

```
<xs:element name="idAttribute" type="idType">
 <xs:annotation>
  <xs:documentation>Language: Reference to ISO 639-2
Other: Reference to the latest version of SutiAttributeList</xs:documentation>
 </xs:annotation>
</xs:element>
```

complexType **attributesType**

diagram



children **attribute**

used by
- elements **addressType/attributeAddress** **content/attributeContent** **driver/attributesDriver** **orderReject/attributesReject** **vehicle/attributesVehicle**

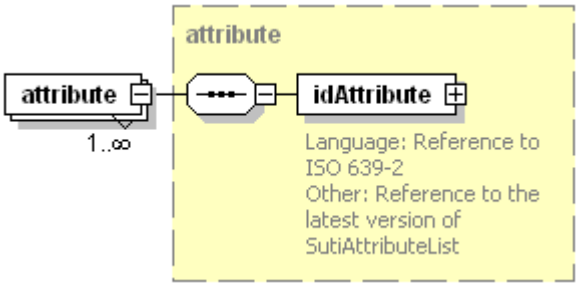
annotation
documentation
A sequence of attributes, where the presence of an attribute means that the existence of the attribute is true

source

```
<xs:complexType name="attributesType">
 <xs:annotation>
  <xs:documentation>A sequence of attributes, where the presence of an attribute means that the
existence of the attribute is true</xs:documentation>
 </xs:annotation>
 <xs:sequence>
  <xs:element name="attribute" type="attribute" maxOccurs="unbounded"/>
 </xs:sequence>
</xs:complexType>
```

element **attributesType/attribute**

diagram



type **attribute**

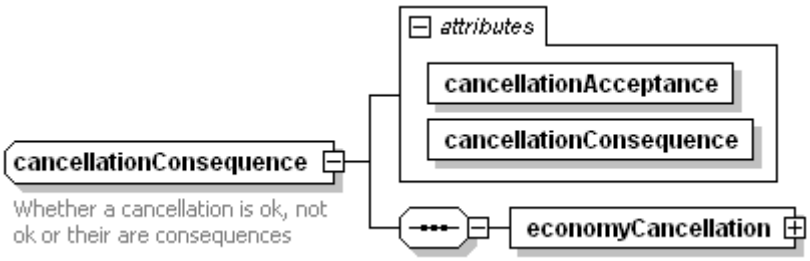
properties
- isRef 0
- minOcc 1
- maxOcc unbounded
- content complex

children **idAttribute**

source
```
<xs:element name="attribute" type="attribute" maxOccurs="unbounded"/>
```

complexType **cancellationConsequence**

diagram



children **economyCancellation**

used by
- element **msg/cancellationConsequence**

attributes

Name	Type	Use	Default	Fixed	annotation
cancellationAcceptance	**xs:boolean**	required			
cancellationConsequence	**xs:boolean**	required			

annotation
documentation
Whether a cancellation is ok, not ok or their are consequences

source
```
<xs:complexType name="cancellationConsequence">
 <xs:annotation>
  <xs:documentation>Whether a cancellation is ok, not ok or their are consequences</xs:documentation>
 </xs:annotation>
 <xs:sequence>
  <xs:element name="economyCancellation" type="economyType"/>
 </xs:sequence>
 <xs:attribute name="cancellationAcceptance" type="xs:boolean" use="required"/>
 <xs:attribute name="cancellationConsequence" type="xs:boolean" use="required"/>
</xs:complexType>
```

attribute **cancellationConsequence/@cancellationAcceptance**

type **xs:boolean**

properties
- isRef 0
- use required

source <xs:attribute name="cancellationAcceptance" type="xs:boolean" use="required"/>

attribute **cancellationConsequence/@cancellationConsequence**

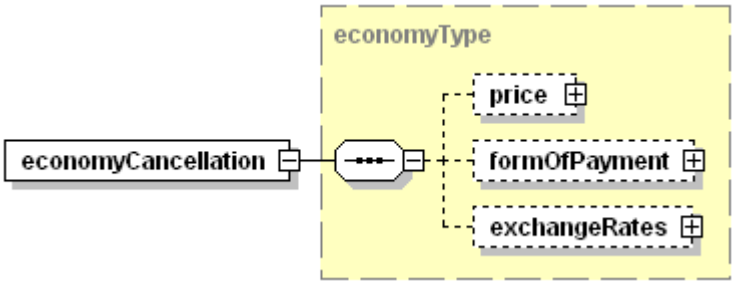
type **xs:boolean**

properties isRef 0
use required

source <xs:attribute name="cancellationConsequence" type="xs:boolean" use="required"/>

element **cancellationConsequence/economyCancellation**

diagram



type **economyType**

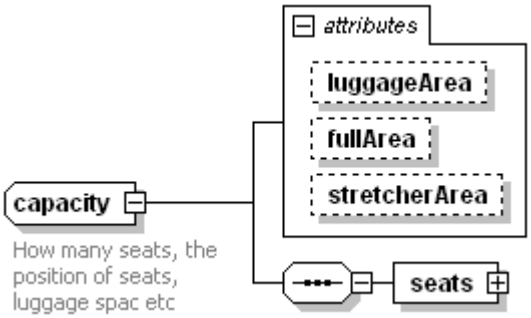
properties isRef 0
content complex

children **price** **formOfPayment** **exchangeRates**

source <xs:element name="economyCancellation" type="economyType"/>

complexType **capacity**

diagram



children **seats**

used by element **vehicle/capacity**

attributes

Name	Type	Use	Default	Fixed	annotation
luggageArea	**xs:float**	optional			
fullArea	**xs:float**	optional			
stretcherArea	**xs:float**	optional			

annotation
documentation
How many seats, the position of seats, luggage spac etc

source
```
<xs:complexType name="capacity">
 <xs:annotation>
  <xs:documentation>How many seats, the position of seats, luggage spac etc</xs:documentation>
 </xs:annotation>
 <xs:sequence>
  <xs:element name="seats" type="seats"/>
 </xs:sequence>
 <xs:attribute name="luggageArea" type="xs:float" use="optional"/>
```

```
  <xs:attribute name="fullArea" type="xs:float" use="optional"/>
  <xs:attribute name="stretcherArea" type="xs:float" use="optional"/>
</xs:complexType>
```

attribute **capacity/@luggageArea**

type **xs:float**

properties
isRef 0
use optional

source
```
<xs:attribute name="luggageArea" type="xs:float" use="optional"/>
```

attribute **capacity/@fullArea**

type **xs:float**

properties
isRef 0
use optional

source
```
<xs:attribute name="fullArea" type="xs:float" use="optional"/>
```

attribute **capacity/@stretcherArea**

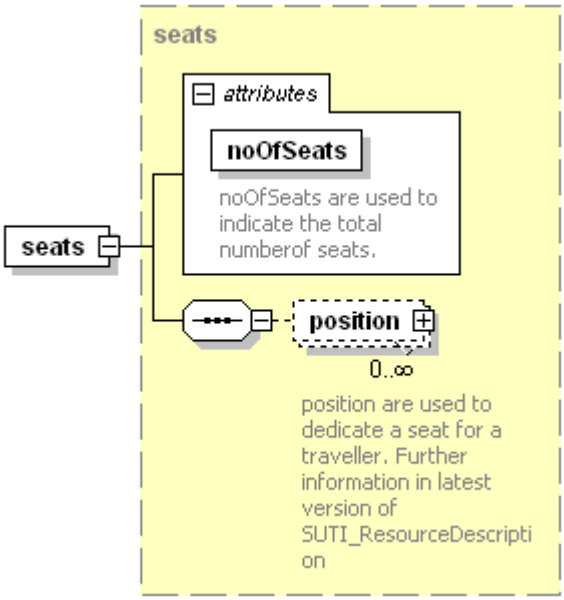
type **xs:float**

properties
isRef 0
use optional

source
```
<xs:attribute name="stretcherArea" type="xs:float" use="optional"/>
```

element **capacity/seats**

diagram



type **seats**

properties
isRef 0
content complex

children **position**

attributes

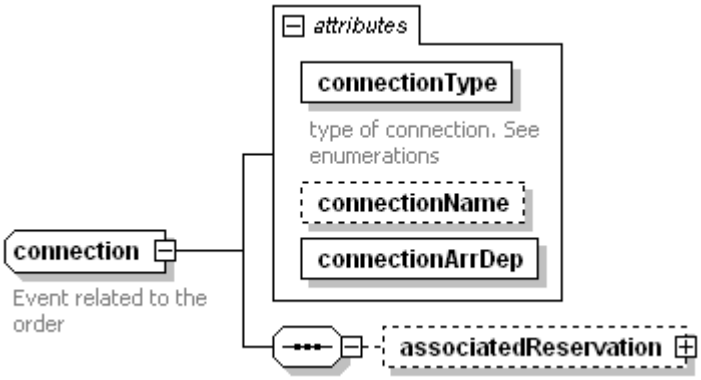
Name	Type	Use	Default	Fixed	annotation
noOfSeats	**xs:nonNegativeInteger**	required			documentation noOfSeats are used to

indicate the total numberof seats.

source
```
<xs:element name="seats" type="seats"/>
```

complexType **connection**

diagram



children **associatedReservation**

used by
element **content/connection**

attributes

Name	Type	Use	Default	Fixed	annotation
connectionType	**derived by: xs:string**	required			documentation type of connection. See enumerations
connectionName	**xs:string**	optional			
connectionArrDep	**xs:string**	required			

annotation
documentation
Event related to the order

source
```
<xs:complexType name="connection">
 <xs:annotation>
  <xs:documentation>Event related to the order</xs:documentation>
 </xs:annotation>
 <xs:sequence>
  <xs:element name="associatedReservation" type="associatedReservation" minOccurs="0"/>
 </xs:sequence>
 <xs:attribute name="connectionType" use="required">
  <xs:annotation>
   <xs:documentation>type of connection. See enumerations</xs:documentation>
  </xs:annotation>
  <xs:simpleType>
   <xs:restriction base="xs:string"/>
  </xs:simpleType>
 </xs:attribute>
 <xs:attribute name="connectionName" type="xs:string" use="optional"/>
 <xs:attribute name="connectionArrDep" type="xs:string" use="required"/>
</xs:complexType>
```

attribute **connection/@connectionType**

type restriction of **xs:string**

properties
isRef 0
use required

annotation
documentation

type of connection. See enumerations

source
```
<xs:attribute name="connectionType" use="required">
 <xs:annotation>
  <xs:documentation>type of connection. See enumerations</xs:documentation>
 </xs:annotation>
 <xs:simpleType>
  <xs:restriction base="xs:string"/>
 </xs:simpleType>
</xs:attribute>
```

attribute **connection/@connectionName**

type **xs:string**

properties
- isRef 0
- use optional

source
```
<xs:attribute name="connectionName" type="xs:string" use="optional"/>
```

attribute **connection/@connectionArrDep**

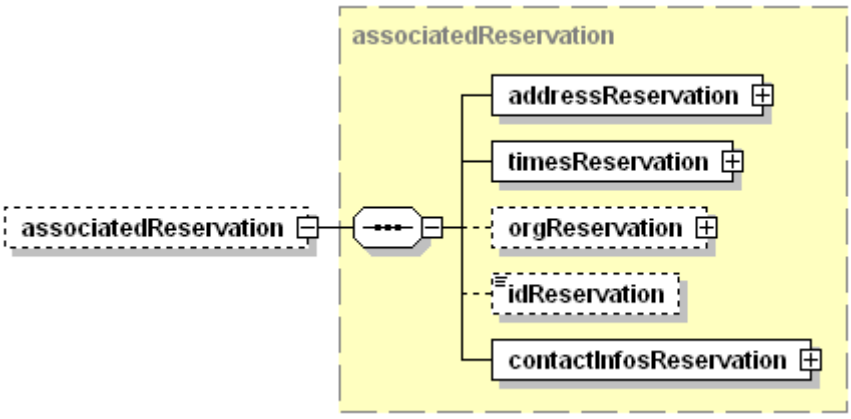
type **xs:string**

properties
- isRef 0
- use required

source
```
<xs:attribute name="connectionArrDep" type="xs:string" use="required"/>
```

element **connection/associatedReservation**

diagram



type **associatedReservation**

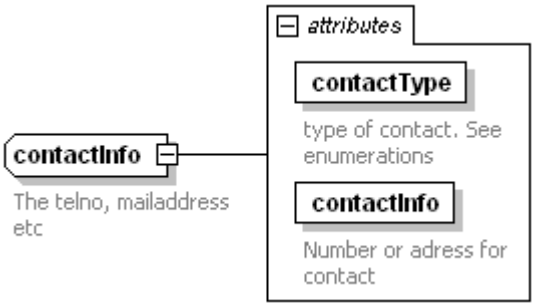
properties
- isRef 0
- minOcc 0
- maxOcc 1
- content complex

children **addressReservation** **timesReservation** **orgReservation** **idReservation** **contactInfosReservation**

source
```
<xs:element name="associatedReservation" type="associatedReservation" minOccurs="0"/>
```

complexType **contactInfo**

diagram



used by element **contactInfosType/contactInfo**

attributes

Name	Type	Use	Default	Fixed	annotation
contactType	**derived by: xs:string**	required			documentation type of contact. See enumerations
contactInfo	**xs:string**	required			documentation Number or adress for contact

annotation
documentation
The telno, mailaddress etc

source

```
<xs:complexType name="contactInfo">
 <xs:annotation>
  <xs:documentation>The telno, mailaddress etc</xs:documentation>
 </xs:annotation>
 <xs:attribute name="contactType" use="required">
  <xs:annotation>
   <xs:documentation>type of contact. See enumerations</xs:documentation>
  </xs:annotation>
  <xs:simpleType>
   <xs:restriction base="xs:string"/>
  </xs:simpleType>
 </xs:attribute>
 <xs:attribute name="contactInfo" type="xs:string" use="required">
  <xs:annotation>
   <xs:documentation>Number or adress for contact</xs:documentation>
  </xs:annotation>
 </xs:attribute>
</xs:complexType>
```

attribute **contactInfo/@contactType**

type restriction of **xs:string**

properties
isRef 0
use required

annotation
documentation
type of contact. See enumerations

source

```
<xs:attribute name="contactType" use="required">
 <xs:annotation>
  <xs:documentation>type of contact. See enumerations</xs:documentation>
 </xs:annotation>
 <xs:simpleType>
  <xs:restriction base="xs:string"/>
 </xs:simpleType>
</xs:attribute>
```

attribute **contactInfo/@contactInfo**

type **xs:string**

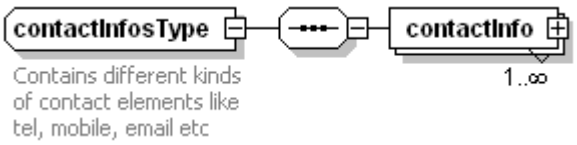
properties
- isRef 0
- use required

annotation
documentation
Number or adress for contact

source
```
<xs:attribute name="contactInfo" type="xs:string" use="required">
 <xs:annotation>
  <xs:documentation>Number or adress for contact</xs:documentation>
 </xs:annotation>
</xs:attribute>
```

complexType **contactInfosType**

diagram



children **contactInfo**

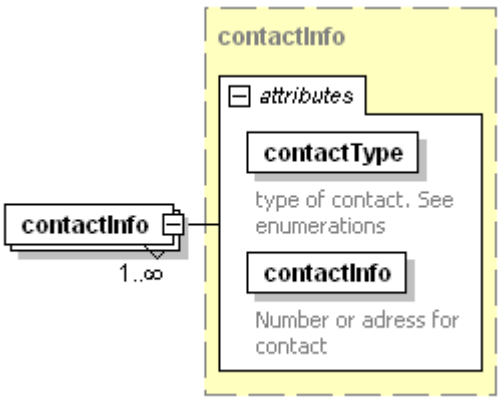
used by
- elements **content/contactInfosContent** **associatedReservation/contactInfosReservation**

annotation
documentation
Contains different kinds of contact elements like tel, mobile, email etc

source
```
<xs:complexType name="contactInfosType">
 <xs:annotation>
  <xs:documentation>Contains different kinds of contact elements like tel, mobile, email
etc</xs:documentation>
 </xs:annotation>
 <xs:sequence>
  <xs:element name="contactInfo" type="contactInfo" maxOccurs="unbounded"/>
 </xs:sequence>
</xs:complexType>
```

element **contactInfosType/contactInfo**

diagram



type **contactInfo**

properties
- isRef 0
- minOcc 1
- maxOcc unbounded
- content complex

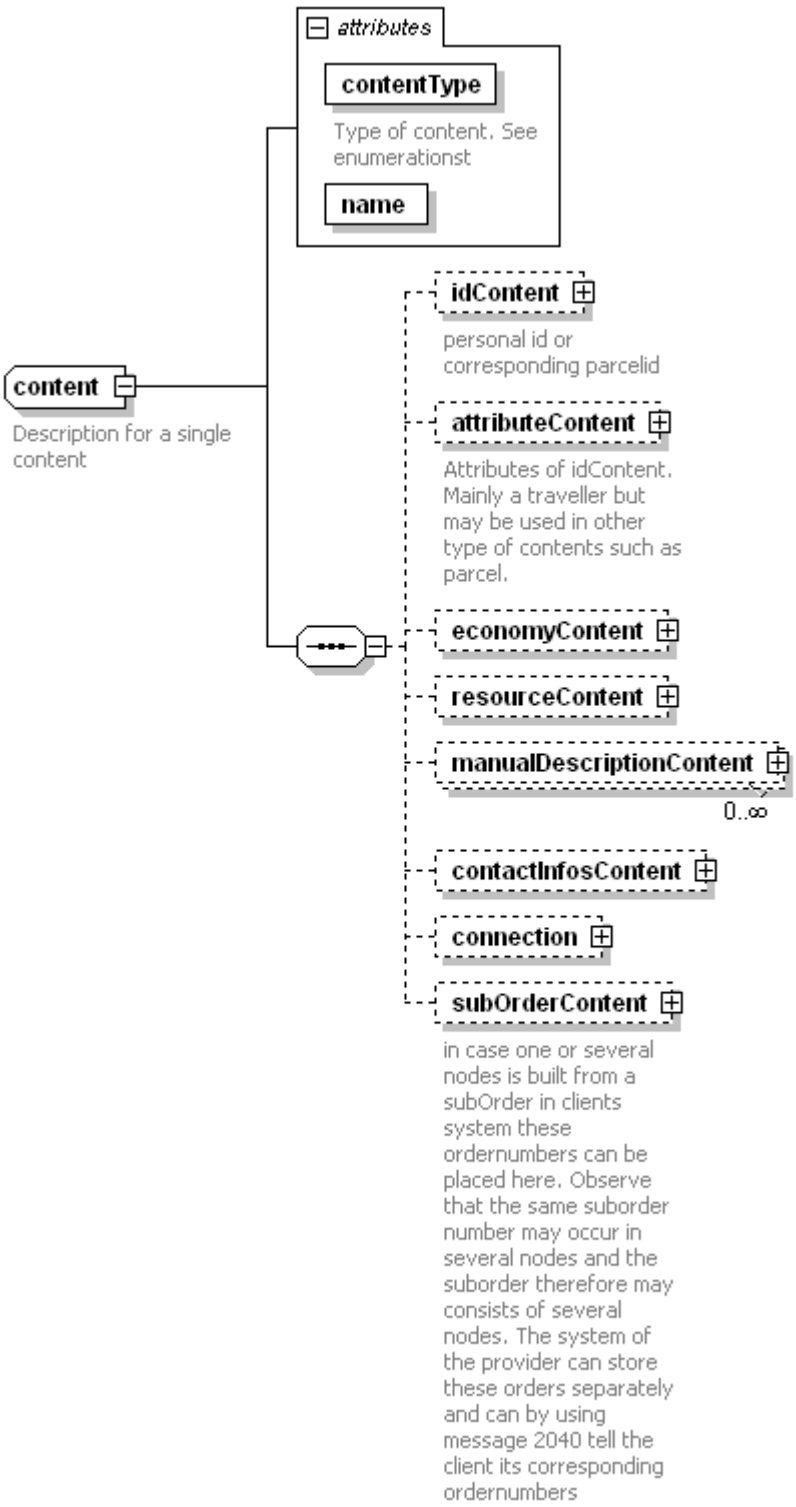
attributes	Name	Type	Use	Default	Fixed	annotation
	contactType	**derived by: xs:string**	required			documentation type of contact. See enumerations
	contactInfo	**xs:string**	required			documentation Number or adress for contact

source

```
<xs:element name="contactInfo" type="contactInfo" maxOccurs="unbounded"/>
```

complexType **content**

diagram



children **idContent** **attributeContent** **economyContent** **resourceContent** **manualDescriptionContent** **contactInfosContent** **connection** **subOrderContent**

used by element **contents/content**

attributes

Name	Type	Use	Default	Fixed	annotation
contentType	**derived by: xs:string**	required			documentation

Type of content. See enumerationst

name **xs:string** required

annotation
documentation
Description for a single content

source

```
<xs:complexType name="content">
 <xs:annotation>
  <xs:documentation>Description for a single content</xs:documentation>
 </xs:annotation>
 <xs:sequence>
  <xs:element name="idContent" minOccurs="0">
   <xs:annotation>
    <xs:documentation>personal id or corresponding parcelid</xs:documentation>
   </xs:annotation>
   <xs:complexType>
    <xs:complexContent>
     <xs:extension base="idType"/>
    </xs:complexContent>
   </xs:complexType>
  </xs:element>
  <xs:element name="attributeContent" type="attributesType" minOccurs="0">
   <xs:annotation>
    <xs:documentation>Attributes of idContent. Mainly a traveller but may be used in other type of
contents such as parcel.</xs:documentation>
   </xs:annotation>
  </xs:element>
  <xs:element name="economyContent" type="economyType" minOccurs="0"/>
  <xs:element name="resourceContent" type="resourceType" minOccurs="0"/>
  <xs:element name="manualDescriptionContent" type="manualDescriptionType" minOccurs="0"
maxOccurs="unbounded"/>
  <xs:element name="contactInfosContent" type="contactInfosType" minOccurs="0"/>
  <xs:element name="connection" type="connection" minOccurs="0"/>
  <xs:element name="subOrderContent" type="subOrderType" minOccurs="0">
   <xs:annotation>
    <xs:documentation>in case one or several nodes is built from a subOrder in clients system these
ordernumbers can be placed here. Observe that the same suborder number may occur in several nodes
and the suborder therefore may consists of several nodes. The system of the provider can store these
orders separately and can by using message 2040 tell the client its corresponding
ordernumbers</xs:documentation>
   </xs:annotation>
  </xs:element>
 </xs:sequence>
 <xs:attribute name="contentType" use="required">
  <xs:annotation>
   <xs:documentation>Type of content. See enumerationst</xs:documentation>
  </xs:annotation>
  <xs:simpleType>
   <xs:restriction base="xs:string"/>
  </xs:simpleType>
 </xs:attribute>
 <xs:attribute name="name" type="xs:string" use="required"/>
</xs:complexType>
```

attribute **content/@contentType**

type restriction of **xs:string**

properties
isRef 0
use required

annotation
documentation
Type of content. See enumerationst

source

```
<xs:attribute name="contentType" use="required">
 <xs:annotation>
  <xs:documentation>Type of content. See enumerationst</xs:documentation>
 </xs:annotation>
```

```
  <xs:simpleType>
    <xs:restriction base="xs:string"/>
  </xs:simpleType>
</xs:attribute>
```

attribute **content/@name**

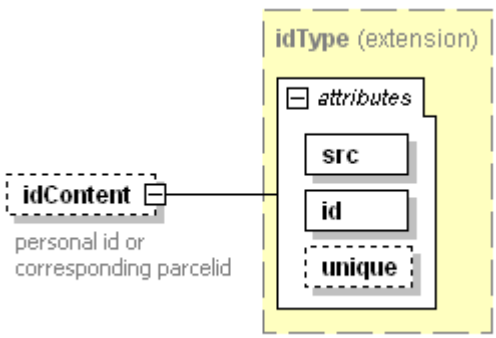
type **xs:string**

properties isRef 0
use required

source
```
<xs:attribute name="name" type="xs:string" use="required"/>
```

element **content/idContent**

diagram



type extension of **idType**

properties isRef 0
minOcc 0
maxOcc 1
content complex

attributes

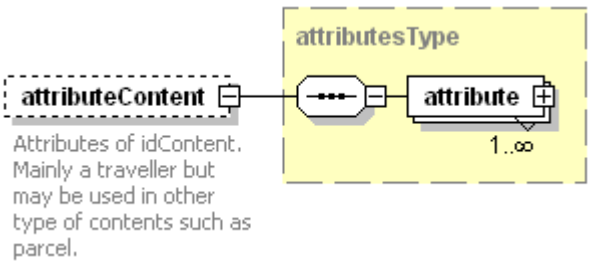
Name	Type	Use	Default	Fixed	annotation
src	**xs:string**	required			
id	**xs:string**	required			
unique	**xs:boolean**	optional	false		

annotation
documentation
personal id or corresponding parcelid

source
```
<xs:element name="idContent" minOccurs="0">
  <xs:annotation>
    <xs:documentation>personal id or corresponding parcelid</xs:documentation>
  </xs:annotation>
  <xs:complexType>
    <xs:complexContent>
      <xs:extension base="idType"/>
    </xs:complexContent>
  </xs:complexType>
</xs:element>
```

element **content/attributeContent**

diagram



type **attributesType**

properties
- isRef 0
- minOcc 0
- maxOcc 1
- content complex

children **attribute**

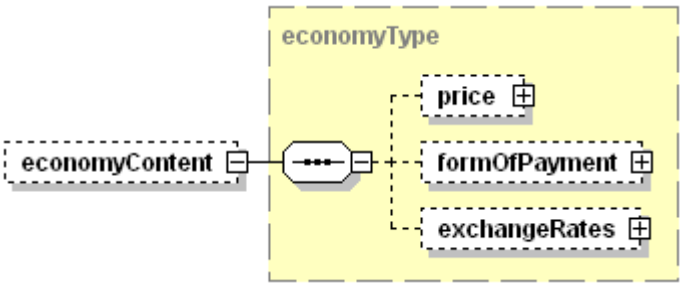
annotation
documentation
Attributes of idContent. Mainly a traveller but may be used in other type of contents such as parcel.

source

```
<xs:element name="attributeContent" type="attributesType" minOccurs="0">
  <xs:annotation>
    <xs:documentation>Attributes of idContent. Mainly a traveller but may be used in other type of contents such as parcel.</xs:documentation>
  </xs:annotation>
</xs:element>
```

element **content/economyContent**

diagram



type **economyType**

properties
- isRef 0
- minOcc 0
- maxOcc 1
- content complex

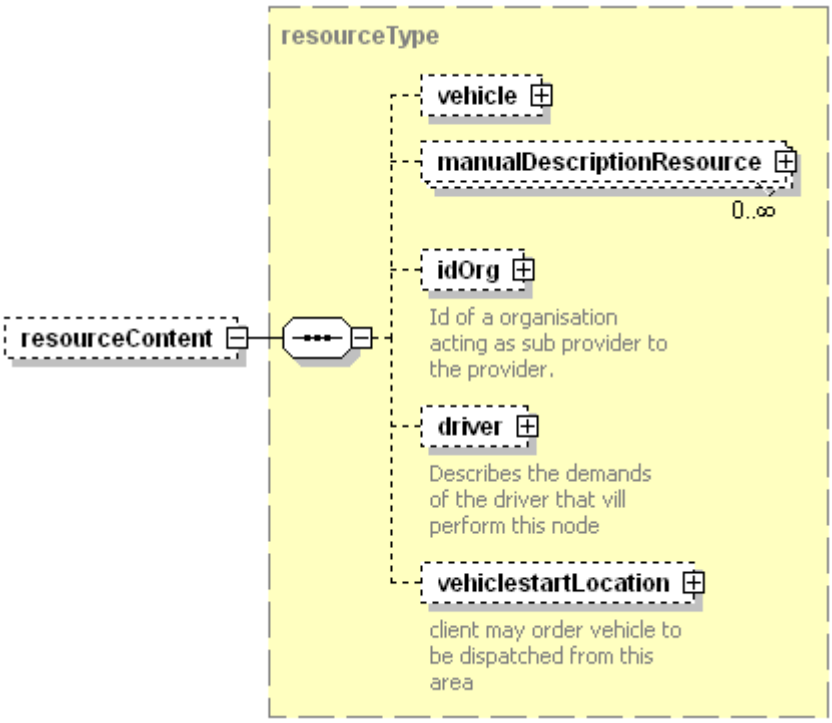
children **price** **formOfPayment** **exchangeRates**

source

```
<xs:element name="economyContent" type="economyType" minOccurs="0"/>
```

element **content/resourceContent**

diagram



type **resourceType**

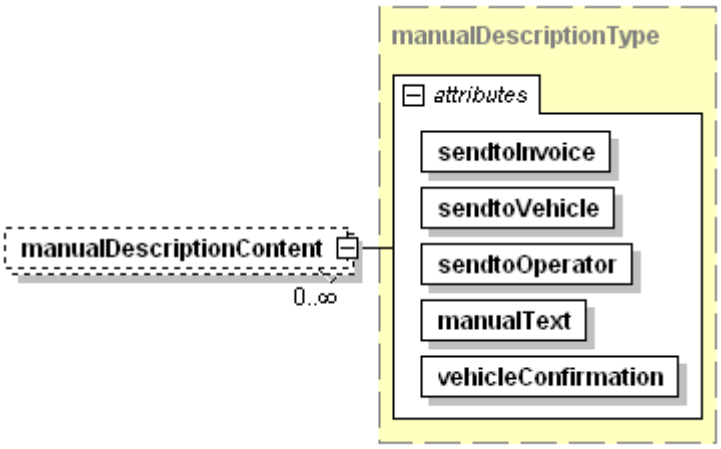
properties
- isRef 0
- minOcc 0
- maxOcc 1
- content complex

children **vehicle** **manualDescriptionResource** **idOrg** **driver** **vehiclestartLocation**

source
```
<xs:element name="resourceContent" type="resourceType" minOccurs="0"/>
```

element **content/manualDescriptionContent**

diagram



type **manualDescriptionType**

properties
- isRef 0
- minOcc 0
- maxOcc unbounded
- content complex

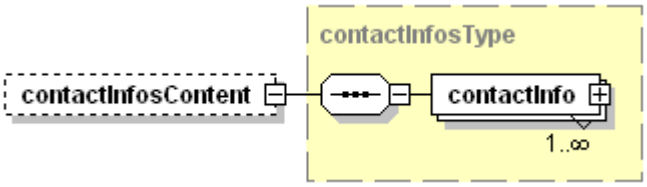
attributes

Name	Type	Use	Default	Fixed	annotation
sendtoInvoice	**xs:boolean**	required			
sendtoVehicle	**xs:boolean**	required			
sendtoOperator	**xs:boolean**	required			
manualText	**xs:string**	required			
vehicleConfirmation	**xs:boolean**	required			

source
```
<xs:element name="manualDescriptionContent" type="manualDescriptionType" minOccurs="0" maxOccurs="unbounded"/>
```

element **content/contactInfosContent**

diagram



type **contactInfosType**

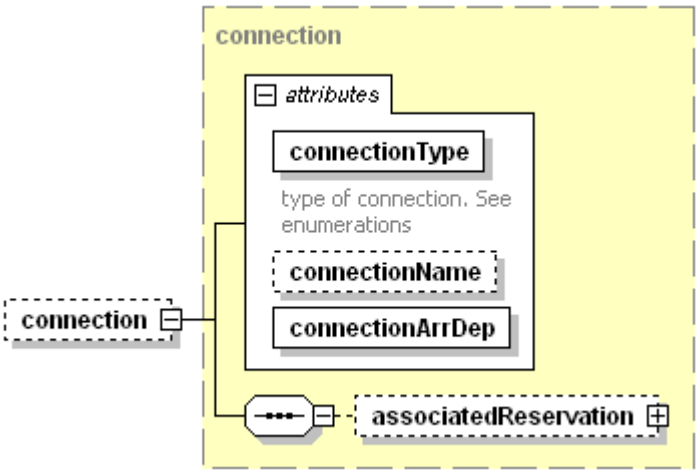
properties
- isRef 0
- minOcc 0
- maxOcc 1
- content complex

children **contactInfo**

source
```
<xs:element name="contactInfosContent" type="contactInfosType" minOccurs="0"/>
```

element **content/connection**

diagram



type **connection**

properties
- isRef 0
- minOcc 0
- maxOcc 1
- content complex

children **associatedReservation**

attributes

Name	Type	Use	Default	Fixed	annotation
connectionType	**derived by: xs:string**	required			documentation: type of connection. See

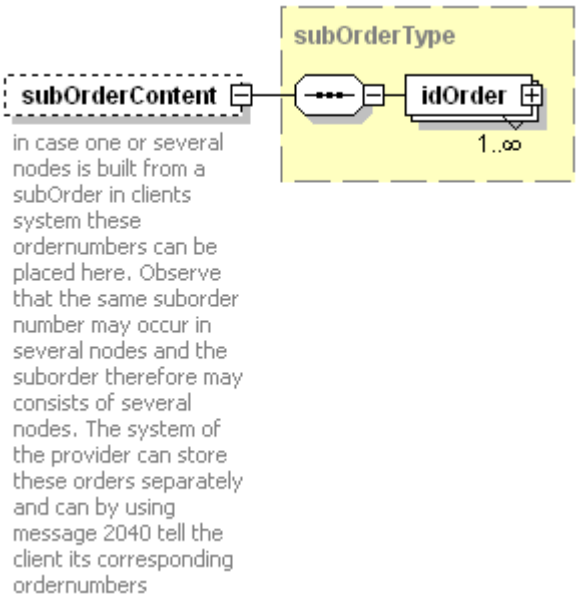
enumerations

connectionName	**xs:string**	optional
connectionArrDep	**xs:string**	required

source
```
<xs:element name="connection" type="connection" minOccurs="0"/>
```

element **content/subOrderContent**

diagram



type **subOrderType**

properties
- isRef 0
- minOcc 0
- maxOcc 1
- content complex

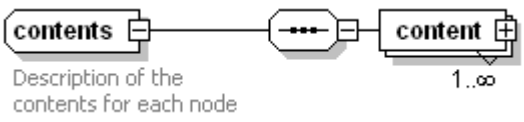
children **idOrder**

annotation
documentation
in case one or several nodes is built from a subOrder in clients system these ordernumbers can be placed here. Observe that the same suborder number may occur in several nodes and the suborder therefore may consists of several nodes. The system of the provider can store these orders separately and can by using message 2040 tell the client its corresponding ordernumbers

source
```
<xs:element name="subOrderContent" type="subOrderType" minOccurs="0">
 <xs:annotation>
  <xs:documentation>in case one or several nodes is built from a subOrder in clients system these ordernumbers can be placed here. Observe that the same suborder number may occur in several nodes and the suborder therefore may consists of several nodes. The system of the provider can store these orders separately and can by using message 2040 tell the client its corresponding ordernumbers</xs:documentation>
 </xs:annotation>
</xs:element>
```

complexType **contents**

diagram



children **content**

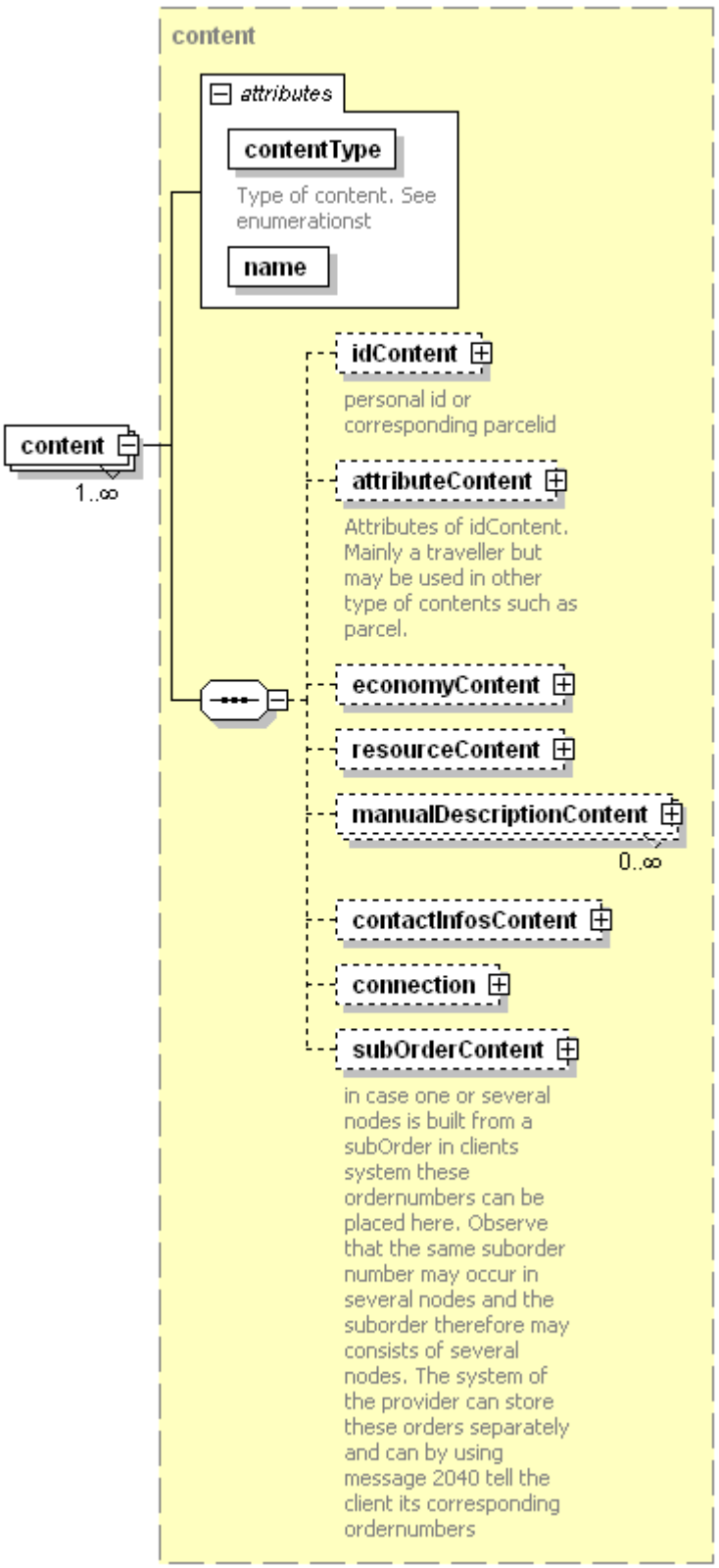
used by element **node/contents**

annotation documentation
Description of the contents for each node

source
```
<xs:complexType name="contents">
 <xs:annotation>
  <xs:documentation>Description of the contents for each node</xs:documentation>
 </xs:annotation>
 <xs:sequence>
  <xs:element name="content" type="content" maxOccurs="unbounded"/>
 </xs:sequence>
</xs:complexType>
```

element **contents/content**

diagram



type **content**

properties

isRef	0
minOcc	1
maxOcc	unbounded

content complex

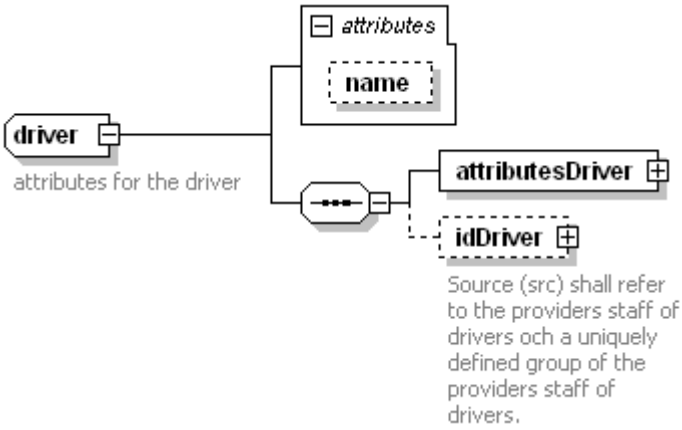
children **idContent attributeContent economyContent resourceContent manualDescriptionContent contactInfosContent connection subOrderContent**

attributes	Name	Type	Use	Default	Fixed	annotation
	contentType	**derived by: xs:string**	required			documentation Type of content. See enumerationst
	name	**xs:string**	required			

source
```
<xs:element name="content" type="content" maxOccurs="unbounded"/>
```

complexType **driver**

diagram



children **attributesDriver idDriver**

used by element **resourceType/driver**

attributes	Name	Type	Use	Default	Fixed	annotation
	name	**xs:string**	optional			

annotation documentation
attributes for the driver

source
```
<xs:complexType name="driver">
 <xs:annotation>
  <xs:documentation>attributes for the driver</xs:documentation>
 </xs:annotation>
 <xs:sequence>
  <xs:element name="attributesDriver" type="attributesType"/>
  <xs:element name="idDriver" type="idType" minOccurs="0">
   <xs:annotation>
    <xs:documentation>Source (src) shall refer to the providers staff of drivers och a uniquely defined group of the providers staff of drivers.</xs:documentation>
   </xs:annotation>
  </xs:element>
 </xs:sequence>
 <xs:attribute name="name" type="xs:string" use="optional"/>
</xs:complexType>
```

attribute **driver/@name**

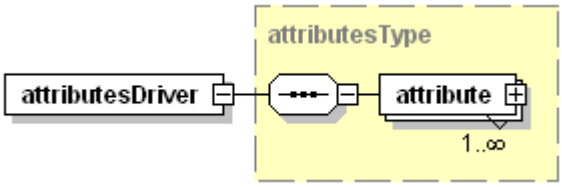
type **xs:string**

properties isRef 0
use optional

source
```
<xs:attribute name="name" type="xs:string" use="optional"/>
```

element **driver/attributesDriver**

diagram



type **attributesType**

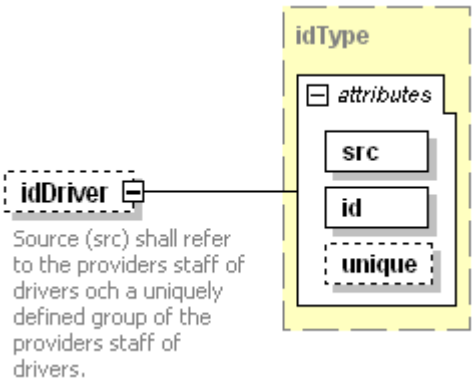
properties
isRef 0
content complex

children **attribute**

source
```
<xs:element name="attributesDriver" type="attributesType"/>
```

element **driver/idDriver**

diagram



type **idType**

properties
isRef 0
minOcc 0
maxOcc 1
content complex

attributes

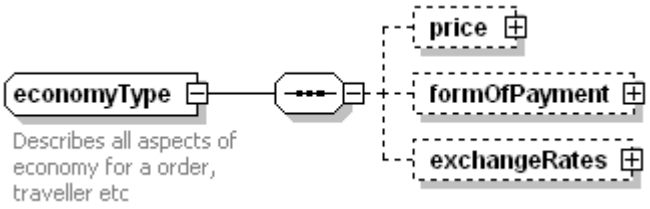
Name	Type	Use	Default	Fixed	annotation
src	**xs:string**	required			
id	**xs:string**	required			
unique	**xs:boolean**	optional	false		

annotation
documentation
Source (src) shall refer to the providers staff of drivers och a uniquely defined group of the providers staff of drivers.

source
```
<xs:element name="idDriver" type="idType" minOccurs="0">
 <xs:annotation>
  <xs:documentation>Source (src) shall refer to the providers staff of drivers och a uniquely defined group of the providers staff of drivers.</xs:documentation>
 </xs:annotation>
</xs:element>
```

complexType **economyType**

diagram



children **price** **formOfPayment** **exchangeRates**

used by elements **cancellationConsequence/economyCancellation** **content/economyContent** **order/economyOrder**

annotation documentation

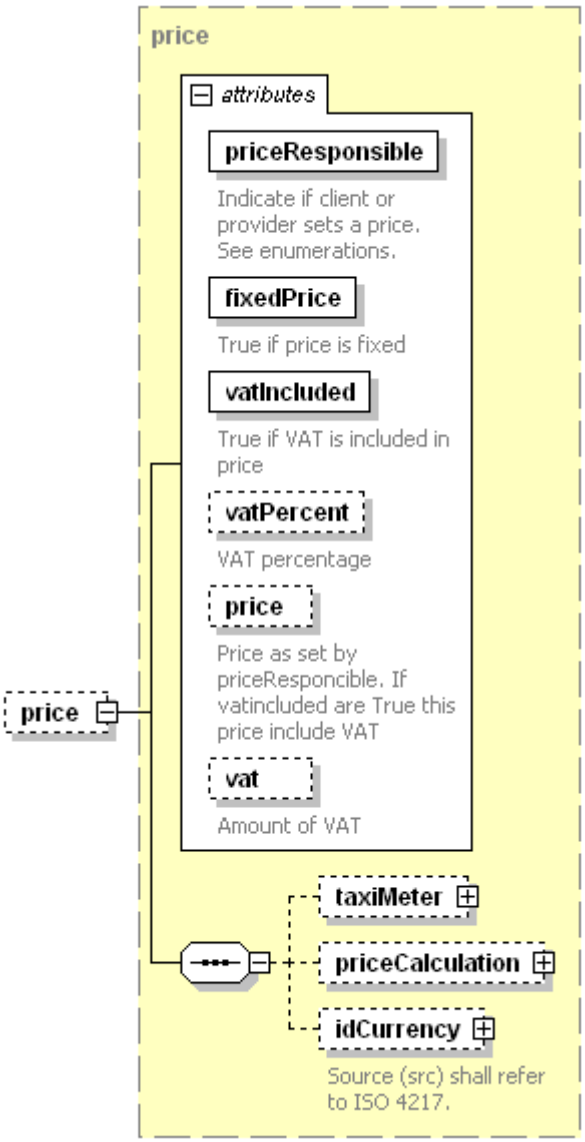
Describes all aspects of economy for a order, traveller etc

source

```
<xs:complexType name="economyType">
  <xs:annotation>
    <xs:documentation>Describes all aspects of economy for a order, traveller etc</xs:documentation>
  </xs:annotation>
  <xs:sequence>
    <xs:element name="price" type="price" minOccurs="0"/>
    <xs:element name="formOfPayment" type="formOfPayment" minOccurs="0"/>
    <xs:element name="exchangeRates" type="exchangeRates" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
```

element **economyType/price**

diagram



type **price**

properties
- isRef 0
- minOcc 0
- maxOcc 1
- content complex

children **taxiMeter** **priceCalculation** **idCurrency**

attributes

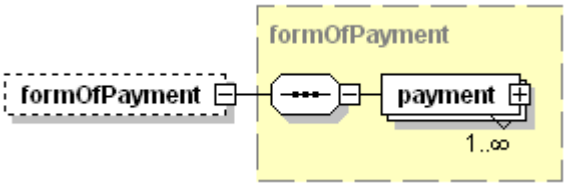
Name	Type	Use	Default	Fixed	annotation
priceResponsible	**derived by: xs:string**	required			documentation Indicate if client or provider sets a price. See enumerations.
fixedPrice	**xs:boolean**	required			documentation True if price is fixed
vatIncluded	**xs:boolean**	required			documentation

vatPercent	**xs:float**	optional	True if VAT is included in price documentation VAT percentage
price	**xs:float**	optional	documentation Price as set by priceResponcible. If vatincluded are True this price include VAT
vat	**xs:float**	optional	documentation Amount of VAT

source `<xs:element name="price" type="price" minOccurs="0"/>`

element **economyType/formOfPayment**

diagram



type **formOfPayment**

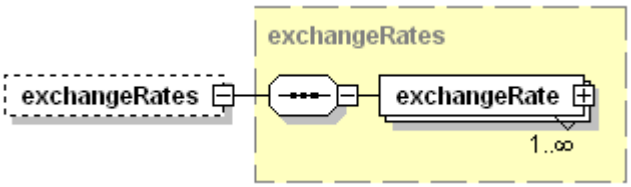
properties
- isRef 0
- minOcc 0
- maxOcc 1
- content complex

children **payment**

source `<xs:element name="formOfPayment" type="formOfPayment" minOccurs="0"/>`

element **economyType/exchangeRates**

diagram



type **exchangeRates**

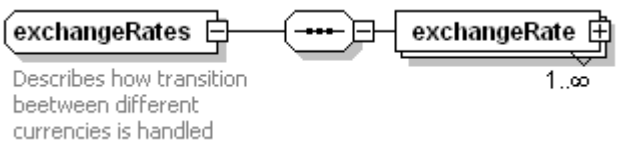
properties
- isRef 0
- minOcc 0
- maxOcc 1
- content complex

children **exchangeRate**

source `<xs:element name="exchangeRates" type="exchangeRates" minOccurs="0"/>`

complexType **exchangeRates**

diagram



children **exchangeRate**

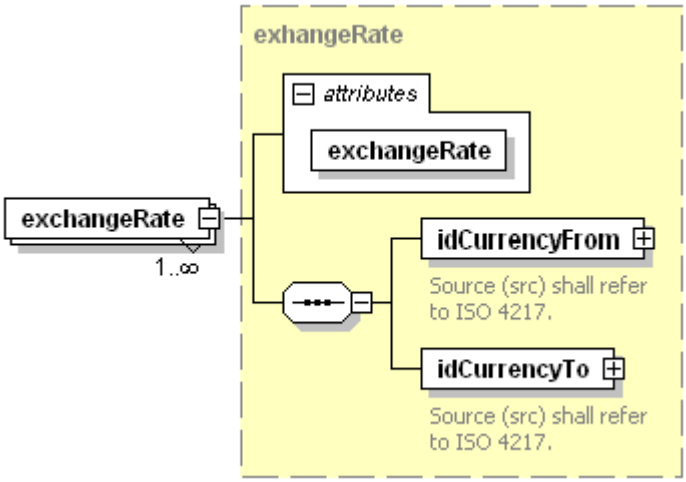
used by element **economyType/exchangeRates**

annotation documentation
Describes how transition beetween different currencies is handled

source
```
<xs:complexType name="exchangeRates">
 <xs:annotation>
  <xs:documentation>Describes how transition beetween different currencies is handled</xs:documentation>
 </xs:annotation>
 <xs:sequence>
  <xs:element name="exchangeRate" type="exhangeRate" maxOccurs="unbounded"/>
 </xs:sequence>
</xs:complexType>
```

element **exchangeRates/exchangeRate**

diagram



type **exhangeRate**

properties
isRef 0
minOcc 1
maxOcc unbounded
content complex

children **idCurrencyFrom** **idCurrencyTo**

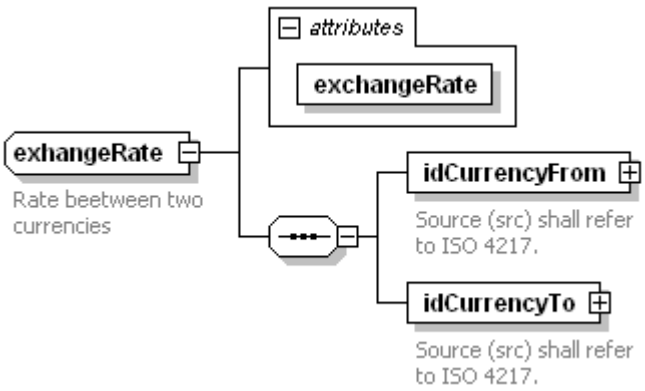
attributes

Name	Type	Use	Default	Fixed	annotation
exchangeRate	**xs:float**	required			

source
```
<xs:element name="exchangeRate" type="exhangeRate" maxOccurs="unbounded"/>
```

complexType **exhangeRate**

diagram



children **idCurrencyFrom** **idCurrencyTo**

used by element **exchangeRates/exchangeRate**

attributes

Name	Type	Use	Default	Fixed	annotation
exchangeRate	**xs:float**	required			

annotation documentation
Rate beetween two currencies

source

```
<xs:complexType name="exhangeRate">
 <xs:annotation>
  <xs:documentation>Rate beetween two currencies</xs:documentation>
 </xs:annotation>
 <xs:sequence>
  <xs:element name="idCurrencyFrom" type="idType">
   <xs:annotation>
    <xs:documentation>Source (src) shall refer to ISO 4217.</xs:documentation>
   </xs:annotation>
  </xs:element>
  <xs:element name="idCurrencyTo" type="idType">
   <xs:annotation>
    <xs:documentation>Source (src) shall refer to ISO 4217.</xs:documentation>
   </xs:annotation>
  </xs:element>
 </xs:sequence>
 <xs:attribute name="exchangeRate" type="xs:float" use="required"/>
</xs:complexType>
```

attribute **exhangeRate/@exchangeRate**

type **xs:float**

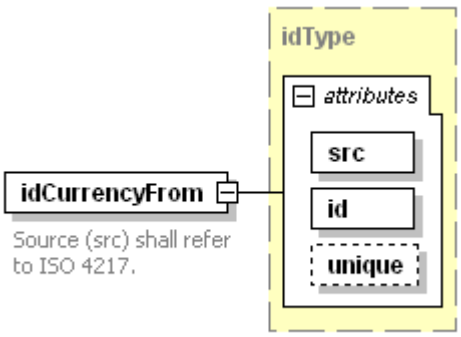
properties
isRef 0
use required

source

```
<xs:attribute name="exchangeRate" type="xs:float" use="required"/>
```

element **exhangeRate/idCurrencyFrom**

diagram



type **idType**

properties
isRef 0
content complex

attributes

Name	Type	Use	Default	Fixed	annotation
src	**xs:string**	required			
id	**xs:string**	required			
unique	**xs:boolean**	optional	false		

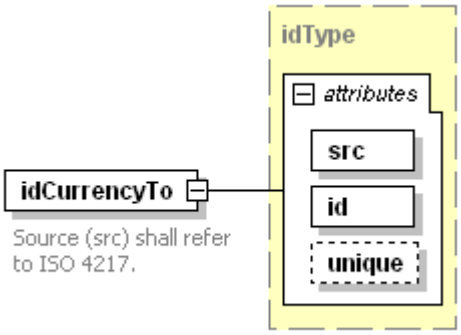
annotation
documentation
Source (src) shall refer to ISO 4217.

source

```
<xs:element name="idCurrencyFrom" type="idType">
 <xs:annotation>
  <xs:documentation>Source (src) shall refer to ISO 4217.</xs:documentation>
 </xs:annotation>
</xs:element>
```

element **exhangeRate/idCurrencyTo**

diagram



type **idType**

properties
isRef 0
content complex

attributes

Name	Type	Use	Default	Fixed	annotation
src	**xs:string**	required			
id	**xs:string**	required			
unique	**xs:boolean**	optional	false		

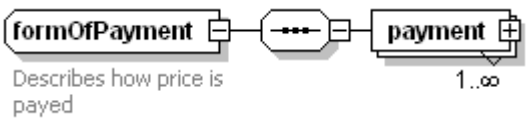
annotation
documentation
Source (src) shall refer to ISO 4217.

source

```
<xs:element name="idCurrencyTo" type="idType">
 <xs:annotation>
  <xs:documentation>Source (src) shall refer to ISO 4217.</xs:documentation>
 </xs:annotation>
</xs:element>
```

complexType **formOfPayment**

diagram



children **payment**

used by
elements **economyType/formOfPayment** **msg/orderReport/economyReport/payment**

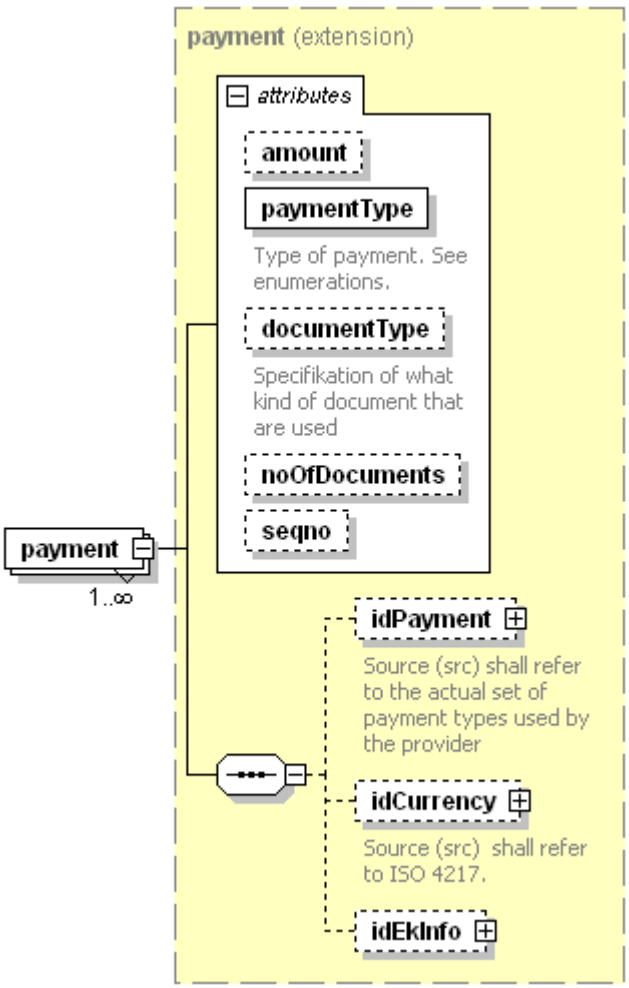
annotation
documentation
Describes how price is payed

source

```
<xs:complexType name="formOfPayment">
 <xs:annotation>
  <xs:documentation>Describes how price is payed</xs:documentation>
 </xs:annotation>
 <xs:sequence>
  <xs:element name="payment" maxOccurs="unbounded">
   <xs:complexType>
    <xs:complexContent>
     <xs:extension base="payment"/>
    </xs:complexContent>
   </xs:complexType>
  </xs:element>
 </xs:sequence>
</xs:complexType>
```

element **formOfPayment/payment**

diagram



type extension of **payment**

properties
- isRef 0
- minOcc 1
- maxOcc unbounded
- content complex

children **idPayment** **idCurrency** **idEkInfo**

attributes

Name	Type	Use	Default	Fixed	annotation
amount	**xs:float**	optional			
paymentType	**derived by: xs:string**	required			documentation Type of payment. See enumerations.
documentType	**xs:string**	optional			documentation Specifikation of what kind of document that are used
noOfDocuments	**xs:nonNegativeInteger**	optional			
seqno	**xs:positiveInteger**	optional			

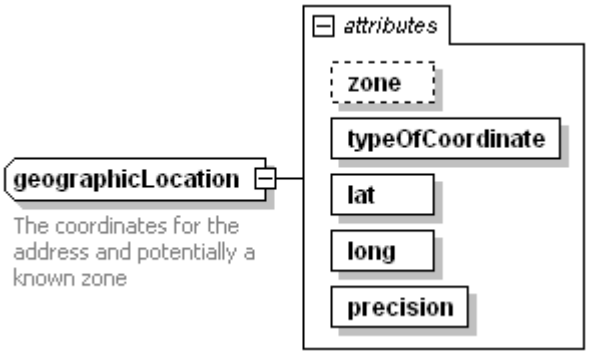
source

```
<xs:element name="payment" maxOccurs="unbounded">
 <xs:complexType>
```

```
    <xs:complexContent>
      <xs:extension base="payment"/>
    </xs:complexContent>
  </xs:complexType>
</xs:element>
```

complexType **geographicLocation**

diagram



used by elements **addressType/geographicLocation** **resourceType/vehiclestartLocation**

attributes

Name	Type	Use	Default	Fixed	annotation
zone	**xs:string**	optional			
typeOfCoordinate	**xs:string**	required			
lat	**xs:float**	required			
long	**xs:float**	required			
precision	**xs:integer**	required			

annotation

documentation

The coordinates for the address and potentially a known zone

source

```
<xs:complexType name="geographicLocation">
  <xs:annotation>
    <xs:documentation>The coordinates for the address and potentially a known zone</xs:documentation>
  </xs:annotation>
  <xs:attribute name="zone" type="xs:string" use="optional"/>
  <xs:attribute name="typeOfCoordinate" type="xs:string" use="required"/>
  <xs:attribute name="lat" type="xs:float" use="required"/>
  <xs:attribute name="long" type="xs:float" use="required"/>
  <xs:attribute name="precision" type="xs:integer" use="required"/>
</xs:complexType>
```

attribute **geographicLocation/@zone**

type **xs:string**

properties

isRef 0
use optional

source

```
<xs:attribute name="zone" type="xs:string" use="optional"/>
```

attribute **geographicLocation/@typeOfCoordinate**

type **xs:string**

properties

isRef 0
use required

source

```
<xs:attribute name="typeOfCoordinate" type="xs:string" use="required"/>
```

attribute **geographicLocation/@lat**

type **xs:float**

properties isRef 0
use required

source
```
<xs:attribute name="lat" type="xs:float" use="required"/>
```

attribute **geographicLocation/@long**

type **xs:float**

properties isRef 0
use required

source
```
<xs:attribute name="long" type="xs:float" use="required"/>
```

attribute **geographicLocation/@precision**

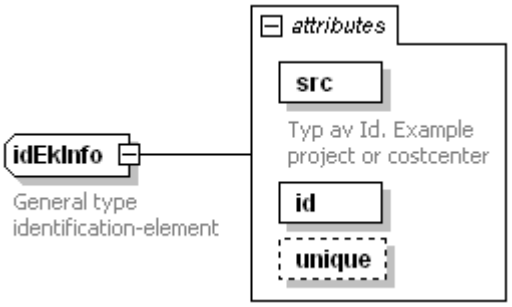
type **xs:integer**

properties isRef 0
use required

source
```
<xs:attribute name="precision" type="xs:integer" use="required"/>
```

complexType **idEkInfo**

diagram



used by element **payment/idEkInfo**

attributes

Name	Type	Use	Default	Fixed	annotation
src	**derived by: xs:string**	required			documentation Typ av Id. Example project or costcenter
id	**xs:string**	required			
unique	**xs:boolean**	optional	false		

annotation
documentation
General type identification-element

source
```
<xs:complexType name="idEkInfo">
 <xs:annotation>
  <xs:documentation>General type identification-element</xs:documentation>
 </xs:annotation>
 <xs:attribute name="src" use="required">
  <xs:annotation>
   <xs:documentation>Typ av Id. Example project or costcenter</xs:documentation>
  </xs:annotation>
  <xs:simpleType>
   <xs:restriction base="xs:string"/>
  </xs:simpleType>
```

```
  </xs:attribute>
  <xs:attribute name="id" type="xs:string" use="required"/>
  <xs:attribute name="unique" type="xs:boolean" use="optional" default="false"/>
 </xs:complexType>
```

attribute **idEkInfo/@src**

type restriction of **xs:string**

properties
isRef 0
use required

annotation
documentation
Typ av Id. Example project or costcenter

source

```
<xs:attribute name="src" use="required">
 <xs:annotation>
  <xs:documentation>Typ av Id. Example project or costcenter</xs:documentation>
 </xs:annotation>
 <xs:simpleType>
  <xs:restriction base="xs:string"/>
 </xs:simpleType>
</xs:attribute>
```

attribute **idEkInfo/@id**

type **xs:string**

properties
isRef 0
use required

source

```
<xs:attribute name="id" type="xs:string" use="required"/>
```

attribute **idEkInfo/@unique**

type **xs:boolean**

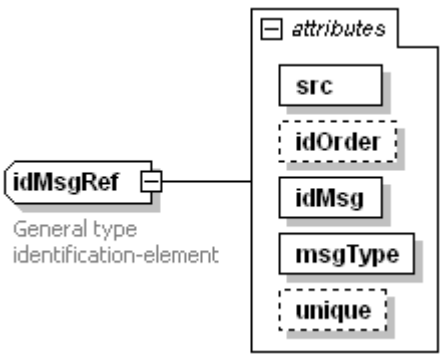
properties
isRef 0
default false
use optional

source

```
<xs:attribute name="unique" type="xs:boolean" use="optional" default="false"/>
```

complexType **idMsgRef**

diagram



used by element **referencesTo/idMsgRef**

attributes

Name	Type	Use	Default	Fixed	annotation
src	**xs:string**	required			
idOrder	**xs:string**				
idMsg	**xs:string**	required			
msgType	**xs:string**	required			
unique	**xs:boolean**	optional	true		

annotation documentation
General type identification-element

source
```
<xs:complexType name="idMsgRef ">
 <xs:annotation>
  <xs:documentation>General type identification-element</xs:documentation>
 </xs:annotation>
 <xs:attribute name="src" type="xs:string" use="required"/>
 <xs:attribute name="idOrder" type="xs:string"/>
 <xs:attribute name="idMsg" type="xs:string" use="required"/>
 <xs:attribute name="msgType" type="xs:string" use="required"/>
 <xs:attribute name="unique" type="xs:boolean" use="optional" default="true"/>
</xs:complexType>
```

attribute **idMsgRef /@src**

type **xs:string**

properties
isRef 0
use required

source
```
<xs:attribute name="src" type="xs:string" use="required"/>
```

attribute **idMsgRef /@idOrder**

type **xs:string**

properties
isRef 0

source
```
<xs:attribute name="idOrder" type="xs:string"/>
```

attribute **idMsgRef /@idMsg**

type **xs:string**

properties
isRef 0
use required

source
```
<xs:attribute name="idMsg" type="xs:string" use="required"/>
```

attribute **idMsgRef /@msgType**

type **xs:string**

properties
isRef 0
use required

source
```
<xs:attribute name="msgType" type="xs:string" use="required"/>
```

attribute **idMsgRef /@unique**

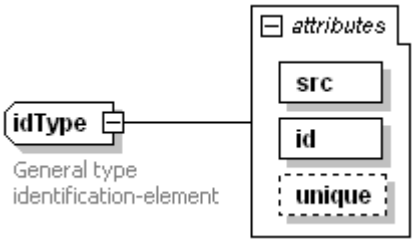
type **xs:boolean**

properties
isRef 0
default true
use optional

source
```
<xs:attribute name="unique" type="xs:boolean" use="optional" default="true"/>
```

complexType **idType**

diagram



used by

elements **addressType/idAddressName** **agreement/idAgreement** **attribute/idAttribute** **addressType/idCommunity** **content/idContent** **addressType/idCountry** **payment/idCurrency** **price/idCurrency** **priceCalculation/idCurrency** **exhangeRate/idCurrencyFrom** **exhangeRate/idCurrencyTo** **driver/idDriver** **referencesTo/idDriver** **addressType/idMap** **referencesTo/idMsg** **orderReject/orderSentBefore/idMsg** **msg/idMsg** **referencesTo/idNode** **subOrderType/idOrder** **order/idOrder** **referencesTo/idOrder** **msg/orderLink/idOrder** **resourceType/idOrg** **orgType/idOrg** **payment/idPayment** **addressType/idPostalCode** **product/idProduct** **addressType/idStreet** **referencesTo/idSuborder** **taxiMeter/idTaxa** **vehicle/idVehicle** **referencesTo/idVehicle** **addressType/idZone**

attributes

Name	Type	Use	Default	Fixed	annotation
src	**xs:string**	required			
id	**xs:string**	required			
unique	**xs:boolean**	optional	false		

annotation

documentation

General type identification-element

source

```
<xs:complexType name="idType">
 <xs:annotation>
  <xs:documentation>General type identification-element</xs:documentation>
 </xs:annotation>
 <xs:attribute name="src" type="xs:string" use="required"/>
 <xs:attribute name="id" type="xs:string" use="required"/>
 <xs:attribute name="unique" type="xs:boolean" use="optional" default="false"/>
</xs:complexType>
```

attribute **idType/@src**

type **xs:string**

properties

isRef 0
use required

source

```
<xs:attribute name="src" type="xs:string" use="required"/>
```

attribute **idType/@id**

type **xs:string**

properties

isRef 0
use required

source

```
<xs:attribute name="id" type="xs:string" use="required"/>
```

attribute **idType/@unique**

type **xs:boolean**

properties

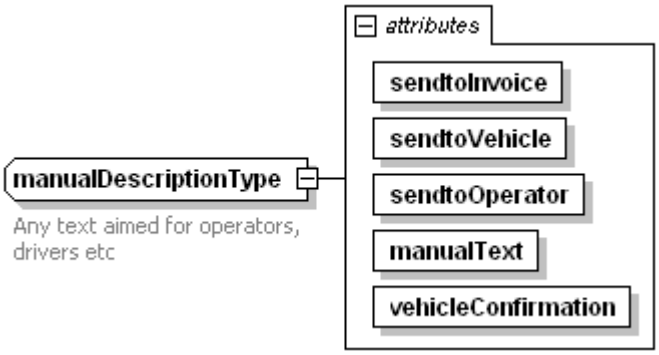
isRef 0
default false
use optional

source

```
<xs:attribute name="unique" type="xs:boolean" use="optional" default="false"/>
```

complexType **manualDescriptionType**

diagram



used by elements **addressType/manualDescriptionAddress** **content/manualDescriptionContent** **msg/manualDescriptionMsg** **resourceType/manualDescriptionResource**

attributes

Name	Type	Use	Default	Fixed	annotation
sendtoInvoice	**xs:boolean**	required			
sendtoVehicle	**xs:boolean**	required			
sendtoOperator	**xs:boolean**	required			
manualText	**xs:string**	required			
vehicleConfirmation	**xs:boolean**	required			

annotation

documentation

Any text aimed for operators, drivers etc

source

```
<xs:complexType name="manualDescriptionType">
 <xs:annotation>
  <xs:documentation>Any text aimed for operators, drivers etc</xs:documentation>
 </xs:annotation>
 <xs:attribute name="sendtoInvoice" type="xs:boolean" use="required"/>
 <xs:attribute name="sendtoVehicle" type="xs:boolean" use="required"/>
 <xs:attribute name="sendtoOperator" type="xs:boolean" use="required"/>
 <xs:attribute name="manualText" type="xs:string" use="required"/>
 <xs:attribute name="vehicleConfirmation" type="xs:boolean" use="required"/>
</xs:complexType>
```

attribute **manualDescriptionType/@sendtoInvoice**

type **xs:boolean**

properties

isRef 0
use required

source

```
<xs:attribute name="sendtoInvoice" type="xs:boolean" use="required"/>
```

attribute **manualDescriptionType/@sendtoVehicle**

type **xs:boolean**

properties

isRef 0
use required

source

```
<xs:attribute name="sendtoVehicle" type="xs:boolean" use="required"/>
```

attribute **manualDescriptionType/@sendtoOperator**

type **xs:boolean**

properties

isRef 0
use required

source

```
<xs:attribute name="sendtoOperator" type="xs:boolean" use="required"/>
```

attribute **manualDescriptionType/@manualText**

type **xs:string**

properties
isRef 0
use required

source
```
<xs:attribute name="manualText" type="xs:string" use="required"/>
```

attribute **manualDescriptionType/@vehicleConfirmation**

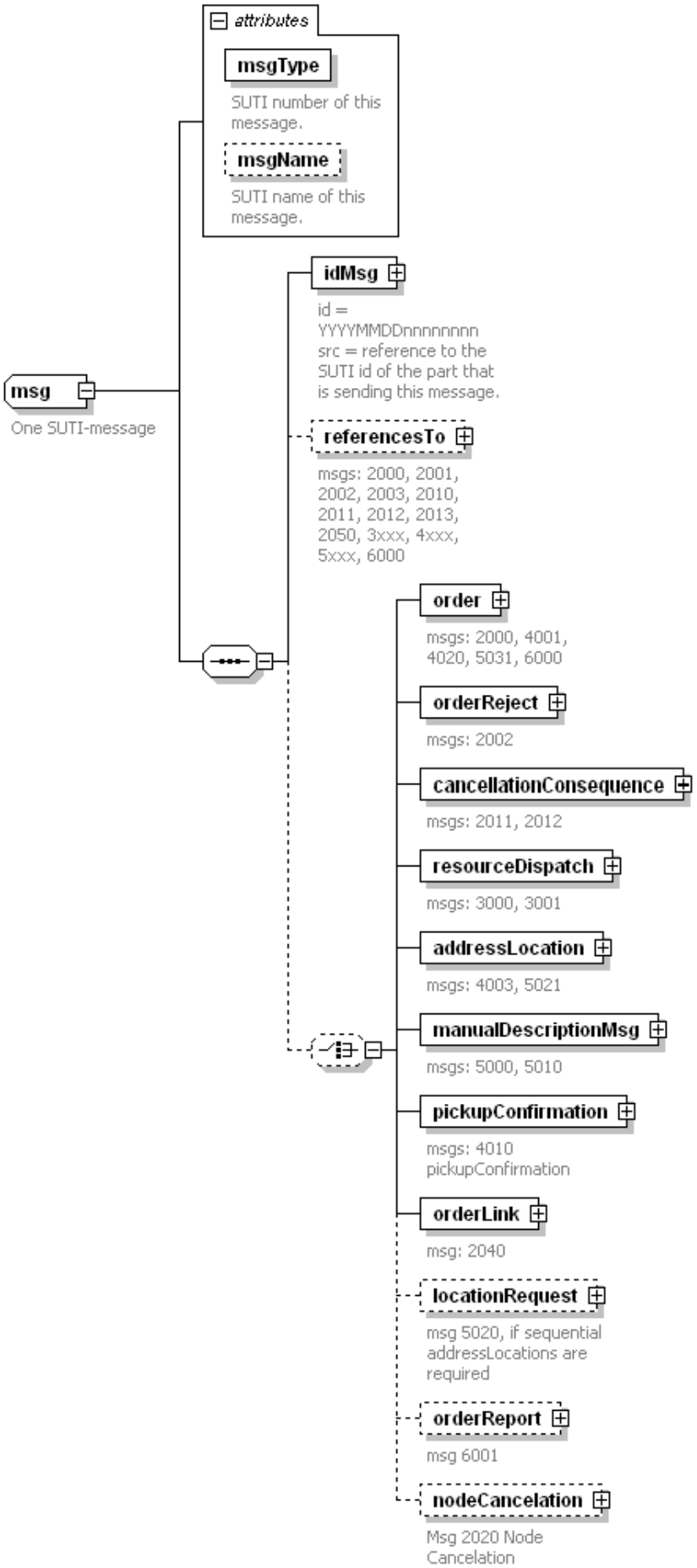
type **xs:boolean**

properties
isRef 0
use required

source
```
<xs:attribute name="vehicleConfirmation" type="xs:boolean" use="required"/>
```

complexType **msg**

diagram


attributes
msgType
SUTI number of this message.
msgName
SUTI name of this message.
msg
One SUTI-message
idMsg
id = YYYYMMDDnnnnnnnn
src = reference to the SUTI id of the part that is sending this message.
referencesTo
msgs: 2000, 2001, 2002, 2003, 2010, 2011, 2012, 2013, 2050, 3xxx, 4xxx, 5xxx, 6000
order
msgs: 2000, 4001, 4020, 5031, 6000
orderReject
msgs: 2002
cancellationConsequence
msgs: 2011, 2012
resourceDispatch
msgs: 3000, 3001
addressLocation
msgs: 4003, 5021
manualDescriptionMsg
msgs: 5000, 5010
pickupConfirmation
msgs: 4010 pickupConfirmation
orderLink
msg: 2040
locationRequest
msg 5020, if sequential addressLocations are required
orderReport
msg 6001
nodeCancelation
Msg 2020 Node Cancelation

children **idMsg** **referencesTo** **order** **orderReject** **cancellationConsequence** **resourceDispatch** **addressLocation** **manualDescriptionMsg** **pickupConfirmation** **orderLink** **locationRequest** **orderReport** **nodeCancelation**

used by element **SUTI/msg**

attributes

Name	Type	Use	Default	Fixed	annotation
msgType	**xs:string**	required			documentation SUTI number of this message.
msgName	**xs:string**	optional			documentation SUTI name of this message.

annotation documentation
One SUTI-message

source

```
<xs:complexType name="msg">
 <xs:annotation>
  <xs:documentation>One SUTI-message</xs:documentation>
 </xs:annotation>
 <xs:sequence>
  <xs:element name="idMsg" type="idType">
   <xs:annotation>
    <xs:documentation>id = YYYYMMDDnnnnnnnn
src = reference to the SUTI id of the part that is sending this message.</xs:documentation>
   </xs:annotation>
  </xs:element>
  <xs:element name="referencesTo" minOccurs="0">
   <xs:annotation>
    <xs:documentation>msgs: 2000, 2001, 2002, 2003, 2010, 2011, 2012, 2013, 2050, 3xxx, 4xxx, 5xxx,
6000</xs:documentation>
   </xs:annotation>
   <xs:complexType>
    <xs:complexContent>
     <xs:extension base="referencesTo"/>
    </xs:complexContent>
   </xs:complexType>
  </xs:element>
  <xs:choice minOccurs="0">
   <xs:element name="order" type="order">
    <xs:annotation>
     <xs:documentation>msgs: 2000, 4001, 4020, 5031, 6000</xs:documentation>
    </xs:annotation>
   </xs:element>
   <xs:element name="orderReject">
    <xs:annotation>
     <xs:documentation>msgs: 2002</xs:documentation>
    </xs:annotation>
    <xs:complexType>
     <xs:complexContent>
      <xs:extension base="orderReject"/>
     </xs:complexContent>
    </xs:complexType>
   </xs:element>
   <xs:element name="cancellationConsequence" type="cancellationConsequence">
    <xs:annotation>
     <xs:documentation>msgs: 2011, 2012</xs:documentation>
    </xs:annotation>
   </xs:element>
   <xs:element name="resourceDispatch" type="resourceType">
    <xs:annotation>
     <xs:documentation>msgs: 3000, 3001</xs:documentation>
    </xs:annotation>
   </xs:element>
   <xs:element name="addressLocation" type="addressType">
```

```
      <xs:annotation>
        <xs:documentation>msgs: 4003, 5021</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="manualDescriptionMsg">
      <xs:annotation>
        <xs:documentation>msgs: 5000, 5010</xs:documentation>
      </xs:annotation>
      <xs:complexType>
        <xs:complexContent>
          <xs:extension base="manualDescriptionType"/>
        </xs:complexContent>
      </xs:complexType>
    </xs:element>
    <xs:element name="pickupConfirmation">
      <xs:annotation>
        <xs:documentation>msgs: 4010 pickupConfirmation</xs:documentation>
      </xs:annotation>
      <xs:complexType>
        <xs:sequence>
          <xs:element name="nodeConfirmed" type="node" minOccurs="0"/>
        </xs:sequence>
        <xs:attribute name="eventType" use="optional">
          <xs:annotation>
            <xs:documentation>Type of event that are beeing confirmed. See
enumerations</xs:documentation>
          </xs:annotation>
          <xs:simpleType>
            <xs:restriction base="xs:string">
              <xs:enumeration value="passengerinvehicle"/>
              <xs:enumeration value="passengerdropped"/>
              <xs:enumeration value="noshow"/>
              <xs:enumeration value="parcelinvehicle"/>
              <xs:enumeration value="parceldropped"/>
              <xs:enumeration value="actiondone"/>
              <xs:enumeration value="navigationdone"/>
            </xs:restriction>
          </xs:simpleType>
        </xs:attribute>
      </xs:complexType>
    </xs:element>
    <xs:element name="orderLink">
      <xs:annotation>
        <xs:documentation>msg: 2040</xs:documentation>
      </xs:annotation>
      <xs:complexType>
        <xs:sequence>
          <xs:element name="idOrder" type="idType" maxOccurs="unbounded">
            <xs:annotation>
              <xs:documentation>ids for the combined order</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="subOrderLink" type="subOrderType" maxOccurs="unbounded">
            <xs:annotation>
              <xs:documentation>ids for all included suborders</xs:documentation>
            </xs:annotation>
          </xs:element>
        </xs:sequence>
      </xs:complexType>
    </xs:element>
    <xs:element name="locationRequest" minOccurs="0">
      <xs:annotation>
        <xs:documentation>msg 5020, if sequential addressLocations are required</xs:documentation>
      </xs:annotation>
      <xs:complexType>
        <xs:sequence>
          <xs:element name="timeFrom" type="time" minOccurs="0">
```

```
      <xs:annotation>
        <xs:documentation>Time to start sending positions to Client</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="timeTo" type="time" minOccurs="0">
      <xs:annotation>
        <xs:documentation>Time to stop sending positions to Client</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="interval" minOccurs="0">
      <xs:annotation>
        <xs:documentation>Interval between positions</xs:documentation>
      </xs:annotation>
      <xs:complexType>
        <xs:attribute name="seconds" type="xs:integer" use="optional"/>
        <xs:attribute name="meter" type="xs:integer" use="optional"/>
      </xs:complexType>
    </xs:element>
  </xs:sequence>
 </xs:complexType>
</xs:element>
<xs:element name="orderReport" minOccurs="0">
 <xs:annotation>
  <xs:documentation>msg 6001</xs:documentation>
 </xs:annotation>
 <xs:complexType>
  <xs:sequence>
   <xs:element name="eventReport" minOccurs="0" maxOccurs="unbounded">
    <xs:complexType>
     <xs:sequence>
      <xs:element name="event" maxOccurs="unbounded">
       <xs:complexType>
        <xs:sequence>
         <xs:element name="eventTime" type="time" minOccurs="0"/>
         <xs:element name="subOrderEvent" type="subOrderType" minOccurs="0"/>
        </xs:sequence>
        <xs:attribute name="nodeSeqno" type="xs:positiveInteger"/>
        <xs:attribute name="eventType" use="required">
         <xs:simpleType>
          <xs:restriction base="xs:string">
           <xs:enumeration value="acceptOrder"/>
           <xs:enumeration value="start"/>
           <xs:enumeration value="stop"/>
           <xs:enumeration value="pickup"/>
           <xs:enumeration value="destination"/>
           <xs:enumeration value="navigation"/>
           <xs:enumeration value="action"/>
          </xs:restriction>
         </xs:simpleType>
        </xs:attribute>
       </xs:complexType>
      </xs:element>
     </xs:sequence>
    </xs:complexType>
   </xs:element>
   <xs:element name="summaryReport" maxOccurs="unbounded">
    <xs:complexType>
     <xs:sequence>
      <xs:element name="subOrderSummary" type="subOrderType" minOccurs="0"/>
     </xs:sequence>
     <xs:attribute name="orderEnded" type="xs:boolean" use="required">
      <xs:annotation>
       <xs:documentation>Indicates that this order is finished.</xs:documentation>
      </xs:annotation>
     </xs:attribute>
     <xs:attribute name="distanceStart" type="xs:integer">
      <xs:annotation>
```

```
            <xs:documentation>Distance from start of order measured in meters</xs:documentation>
           </xs:annotation>
          </xs:attribute>
          <xs:attribute name="durationStart" type="xs:integer">
           <xs:annotation>
            <xs:documentation>Duration from start of order measured in seconds</xs:documentation>
           </xs:annotation>
          </xs:attribute>
          <xs:attribute name="vehicle" type="xs:string">
           <xs:annotation>
            <xs:documentation>Vehicle that has performed this order</xs:documentation>
           </xs:annotation>
          </xs:attribute>
         </xs:complexType>
        </xs:element>
        <xs:element name="economyReport" minOccurs="0" maxOccurs="unbounded">
         <xs:annotation>
          <xs:documentation>If payment are separated for each node or suborder one economyreport for
each node or suborder can be sent</xs:documentation>
         </xs:annotation>
         <xs:complexType>
          <xs:sequence>
           <xs:element name="payment" type="formOfPayment"/>
           <xs:element name="subOrderEconomy" type="subOrderType" minOccurs="0">
            <xs:annotation>
             <xs:documentation>Used if economyreport are sent for each suborder</xs:documentation>
            </xs:annotation>
           </xs:element>
          </xs:sequence>
          <xs:attribute name="nodeSeqno" type="xs:positiveInteger">
           <xs:annotation>
            <xs:documentation>Used if economyreport are sent for each node</xs:documentation>
           </xs:annotation>
          </xs:attribute>
         </xs:complexType>
        </xs:element>
       </xs:sequence>
      </xs:complexType>
     </xs:element>
     <xs:element name="nodeCancelation" minOccurs="0">
      <xs:annotation>
       <xs:documentation>Msg 2020 Node Cancelation</xs:documentation>
      </xs:annotation>
      <xs:complexType>
       <xs:attribute name="nodeStart" type="xs:positiveInteger" use="required">
        <xs:annotation>
         <xs:documentation>The startnode that shall be cancelled. Often a pickup</xs:documentation>
        </xs:annotation>
       </xs:attribute>
       <xs:attribute name="nodeEnd" type="xs:positiveInteger">
        <xs:annotation>
         <xs:documentation>The endnode that shall be cancelled. Often a destination</xs:documentation>
        </xs:annotation>
       </xs:attribute>
      </xs:complexType>
     </xs:element>
    </xs:choice>
   </xs:sequence>
   <xs:attribute name="msgType" type="xs:string" use="required">
    <xs:annotation>
     <xs:documentation>SUTI number of this message.</xs:documentation>
    </xs:annotation>
   </xs:attribute>
   <xs:attribute name="msgName" type="xs:string" use="optional">
    <xs:annotation>
     <xs:documentation>SUTI name of this message.</xs:documentation>
    </xs:annotation>
```

```
  </xs:attribute>
</xs:complexType>
```

attribute **msg/@msgType**

type **xs:string**

properties
isRef 0
use required

annotation
documentation
SUTI number of this message.

source
```
<xs:attribute name="msgType" type="xs:string" use="required">
  <xs:annotation>
    <xs:documentation>SUTI number of this message.</xs:documentation>
  </xs:annotation>
</xs:attribute>
```

attribute **msg/@msgName**

type **xs:string**

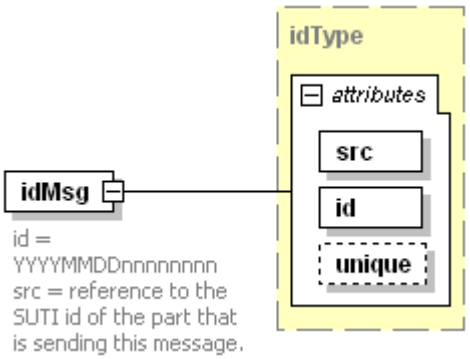
properties
isRef 0
use optional

annotation
documentation
SUTI name of this message.

source
```
<xs:attribute name="msgName" type="xs:string" use="optional">
  <xs:annotation>
    <xs:documentation>SUTI name of this message.</xs:documentation>
  </xs:annotation>
</xs:attribute>
```

element **msg/idMsg**

diagram



type **idType**

properties
isRef 0
content complex

attributes

Name	Type	Use	Default	Fixed	annotation
src	**xs:string**	required			
id	**xs:string**	required			
unique	**xs:boolean**	optional	false		

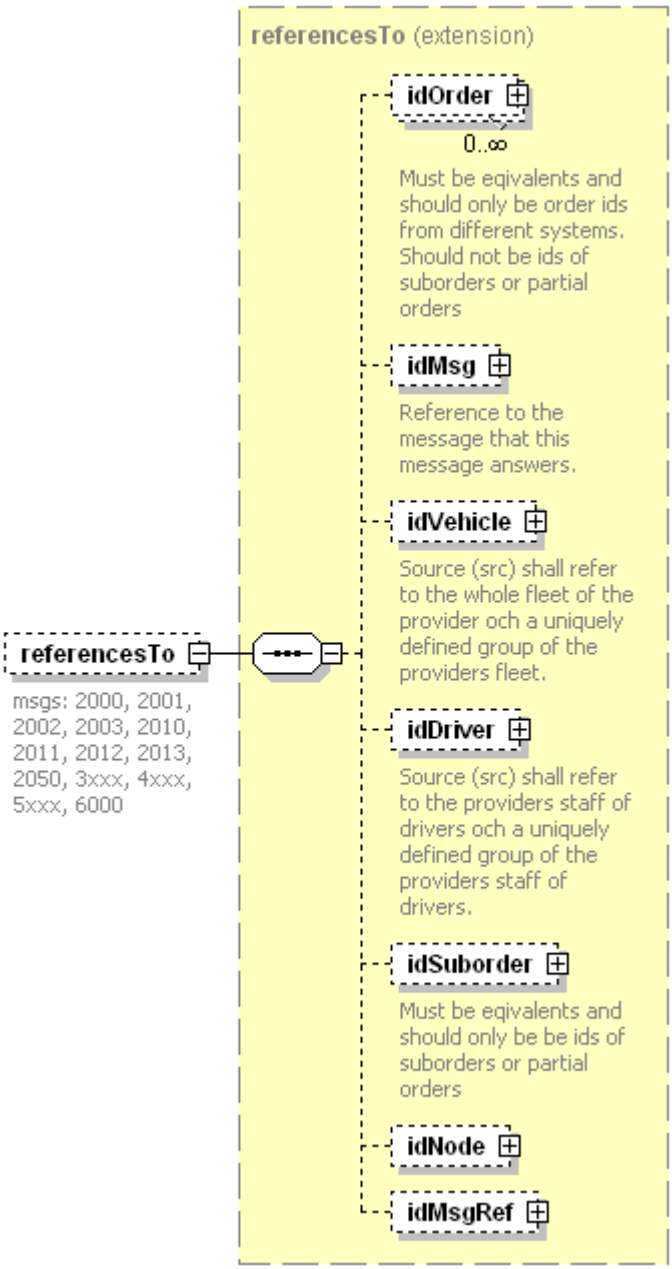
annotation
documentation
id = YYYYMMDDnnnnnnnn
src = reference to the SUTI id of the part that is sending this message.

source
```
<xs:element name="idMsg" type="idType">
  <xs:annotation>
    <xs:documentation>id = YYYYMMDDnnnnnnnn
src = reference to the SUTI id of the part that is sending this message.</xs:documentation>
  </xs:annotation>
```

```
</xs:element>
```

element **msg/referencesTo**

diagram



type extension of **referencesTo**

properties

isRef	0
minOcc	0
maxOcc	1
content	complex

children **idOrder** **idMsg** **idVehicle** **idDriver** **idSuborder** **idNode** **idMsgRef**

annotation
documentation
msgs: 2000, 2001, 2002, 2003, 2010, 2011, 2012, 2013, 2050, 3xxx, 4xxx, 5xxx, 6000

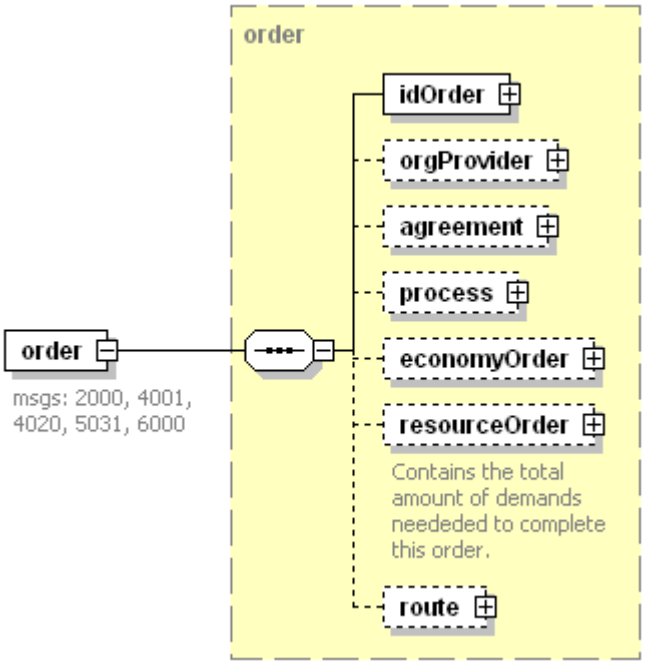
source

```
<xs:element name="referencesTo" minOccurs="0">
  <xs:annotation>
    <xs:documentation>msgs: 2000, 2001, 2002, 2003, 2010, 2011, 2012, 2013, 2050, 3xxx, 4xxx, 5xxx, 6000</xs:documentation>
  </xs:annotation>
```

```
  <xs:complexType>
   <xs:complexContent>
    <xs:extension base="referencesTo"/>
   </xs:complexContent>
  </xs:complexType>
</xs:element>
```

element **msg/order**

diagram



type **order**

properties
isRef 0
content complex

children **idOrder** **orgProvider** **agreement** **process** **economyOrder** **resourceOrder** **route**

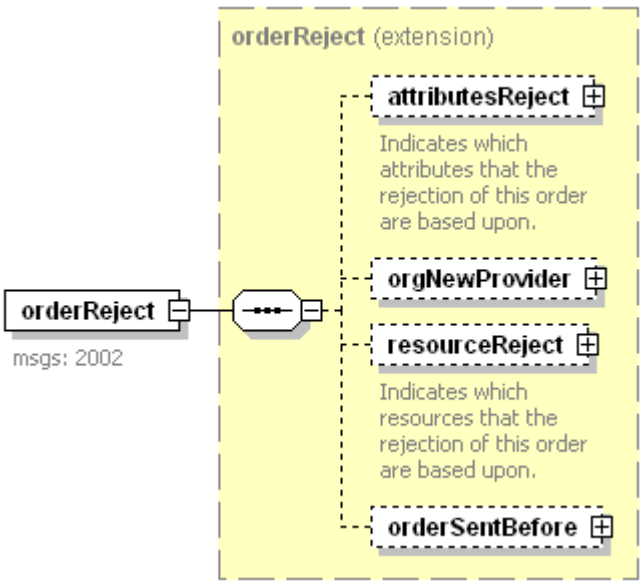
annotation
documentation
msgs: 2000, 4001, 4020, 5031, 6000

source

```
<xs:element name="order" type="order">
 <xs:annotation>
  <xs:documentation>msgs: 2000, 4001, 4020, 5031, 6000</xs:documentation>
 </xs:annotation>
</xs:element>
```

element **msg/orderReject**

diagram



type extension of **orderReject**

properties
- isRef 0
- content complex

children **attributesReject** **orgNewProvider** **resourceReject** **orderSentBefore**

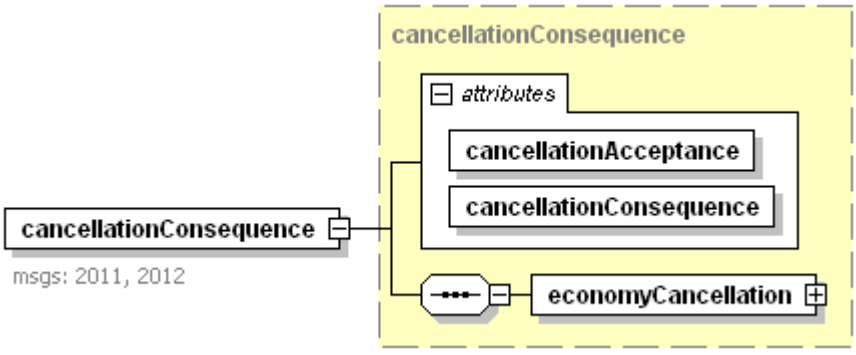
annotation
documentation
msgs: 2002

source

```
<xs:element name="orderReject">
 <xs:annotation>
  <xs:documentation>msgs: 2002</xs:documentation>
 </xs:annotation>
 <xs:complexType>
  <xs:complexContent>
   <xs:extension base="orderReject"/>
  </xs:complexContent>
 </xs:complexType>
</xs:element>
```

element **msg/cancellationConsequence**

diagram



type **cancellationConsequence**

properties
- isRef 0
- content complex

children **economyCancellation**

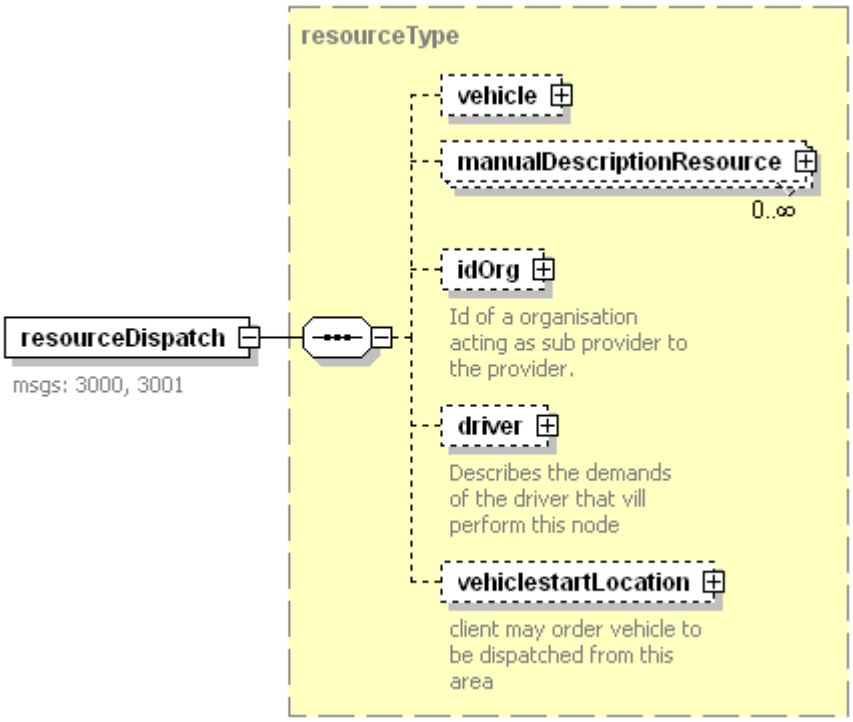
attributes	Name	Type	Use	Default	Fixed	annotation
	cancellationAcceptance	**xs:boolean**	required			
	cancellationConsequence	**xs:boolean**	required			

annotation
documentation
msgs: 2011, 2012

source
```
<xs:element name="cancellationConsequence" type="cancellationConsequence">
 <xs:annotation>
  <xs:documentation>msgs: 2011, 2012</xs:documentation>
 </xs:annotation>
</xs:element>
```

element **msg/resourceDispatch**

diagram



type **resourceType**

properties
isRef 0
content complex

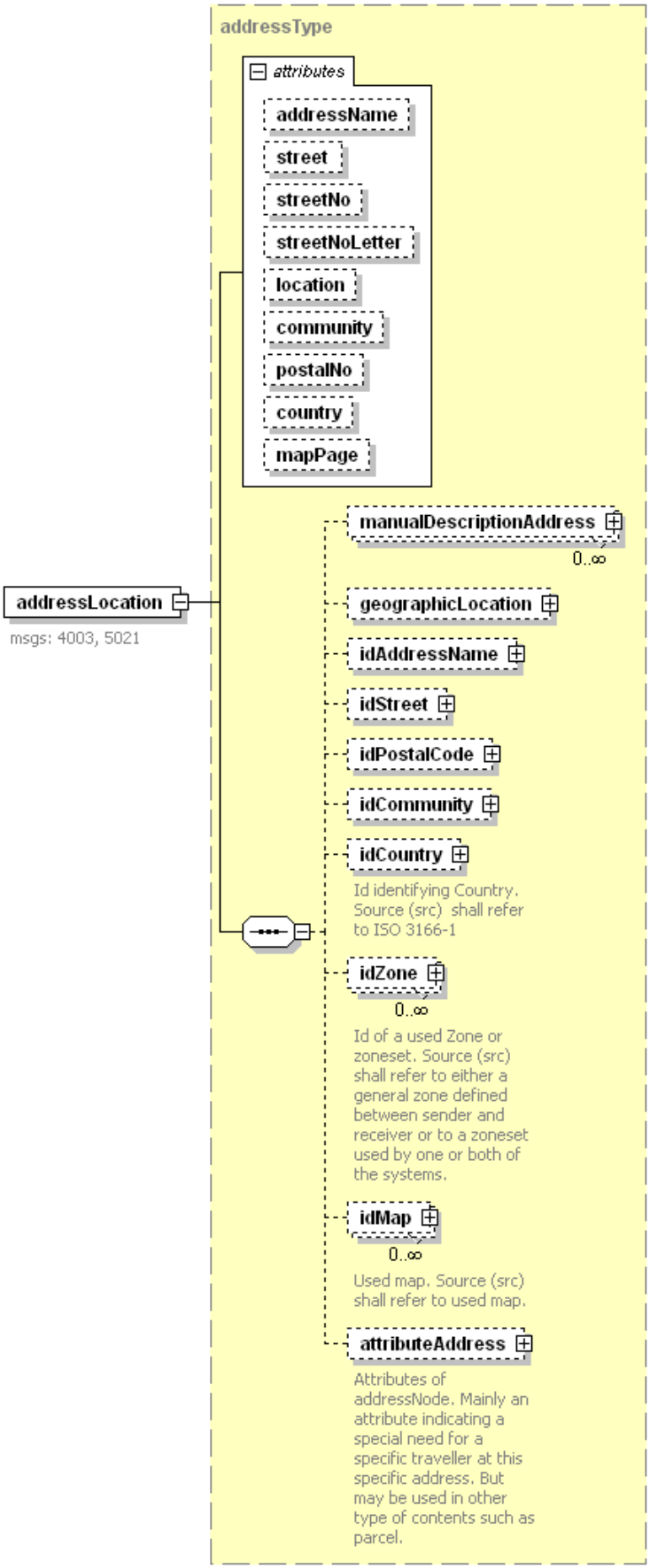
children **vehicle** **manualDescriptionResource** **idOrg** **driver** **vehiclestartLocation**

annotation
documentation
msgs: 3000, 3001

source
```
<xs:element name="resourceDispatch" type="resourceType">
 <xs:annotation>
  <xs:documentation>msgs: 3000, 3001</xs:documentation>
 </xs:annotation>
</xs:element>
```

element **msg/addressLocation**

diagram


addressType
attributes
addressName
street
streetNo
streetNoLetter
location
community
postalNo
country
mapPage
addressLocation
msgs: 4003, 5021
manualDescriptionAddress
0..∞
geographicLocation
idAddressName
idStreet
idPostalCode
idCommunity
idCountry
Id identifying Country. Source (src) shall refer to ISO 3166-1
idZone
0..∞
Id of a used Zone or zoneset. Source (src) shall refer to either a general zone defined between sender and receiver or to a zoneset used by one or both of the systems.
idMap
0..∞
Used map. Source (src) shall refer to used map.
attributeAddress
Attributes of addressNode. Mainly an attribute indicating a special need for a specific traveller at this specific address. But may be used in other type of contents such as parcel.

type **addressType**

properties
isRef 0
content complex

children **manualDescriptionAddress** **geographicLocation** **idAddressName** **idStreet** **idPostalCode** **idCommunity** **idCountry** **idZone** **idMap** **attributeAddress**

attributes

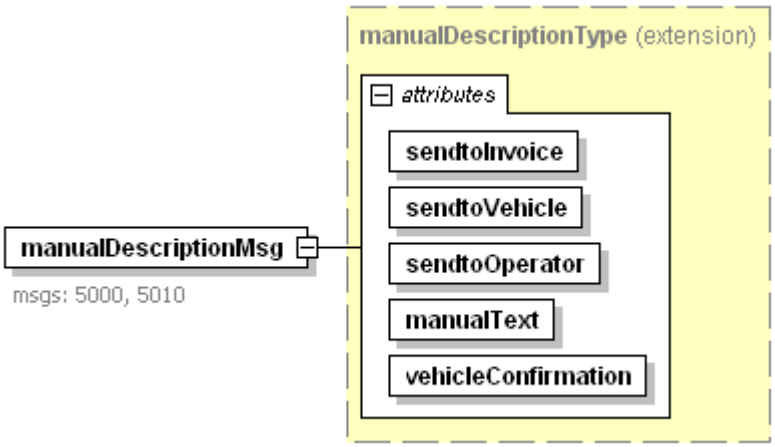
Name	Type	Use	Default	Fixed	annotation
addressName	**xs:string**	optional			
street	**xs:string**	optional			
streetNo	**xs:positiveInteger**	optional			
streetNoLetter	**xs:string**	optional			
location	**xs:string**	optional			
community	**xs:string**	optional			
postalNo	**xs:string**	optional			
country	**xs:string**	optional			
mapPage	**xs:string**	optional			

annotation
documentation
msgs: 4003, 5021

source
```
<xs:element name="addressLocation" type="addressType">
 <xs:annotation>
  <xs:documentation>msgs: 4003, 5021</xs:documentation>
 </xs:annotation>
</xs:element>
```

element **msg/manualDescriptionMsg**

diagram



type extension of **manualDescriptionType**

properties
isRef 0
content complex

attributes

Name	Type	Use	Default	Fixed	annotation
sendtoInvoice	**xs:boolean**	required			
sendtoVehicle	**xs:boolean**	required			
sendtoOperator	**xs:boolean**	required			
manualText	**xs:string**	required			
vehicleConfirmation	**xs:boolean**	required			

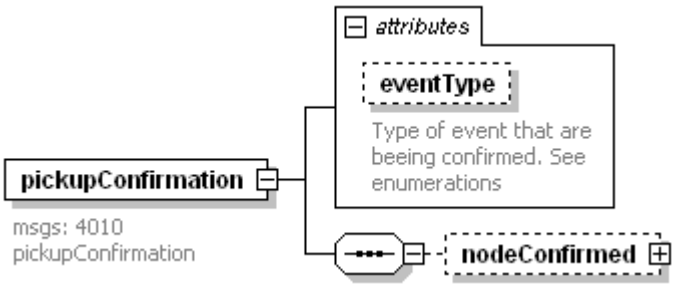
annotation
documentation
msgs: 5000, 5010

source
```
<xs:element name="manualDescriptionMsg">
 <xs:annotation>
  <xs:documentation>msgs: 5000, 5010</xs:documentation>
 </xs:annotation>
 <xs:complexType>
  <xs:complexContent>
```

```
    <xs:extension base="manualDescriptionType"/>
   </xs:complexContent>
  </xs:complexType>
 </xs:element>
```

element **msg/pickupConfirmation**

diagram



properties
- isRef 0
- content complex

children **nodeConfirmed**

attributes

Name	Type	Use	Default	Fixed	annotation
eventType	**derived by: xs:string**	optional			documentation: Type of event that are beeing confirmed. See enumerations

annotation
documentation
msgs: 4010 pickupConfirmation

source

```
<xs:element name="pickupConfirmation">
 <xs:annotation>
  <xs:documentation>msgs: 4010 pickupConfirmation</xs:documentation>
 </xs:annotation>
 <xs:complexType>
  <xs:sequence>
   <xs:element name="nodeConfirmed" type="node" minOccurs="0"/>
  </xs:sequence>
  <xs:attribute name="eventType" use="optional">
   <xs:annotation>
    <xs:documentation>Type of event that are beeing confirmed. See enumerations</xs:documentation>
   </xs:annotation>
   <xs:simpleType>
    <xs:restriction base="xs:string">
     <xs:enumeration value="passengerinvehicle"/>
     <xs:enumeration value="passengerdropped"/>
     <xs:enumeration value="noshow"/>
     <xs:enumeration value="parcelinvehicle"/>
     <xs:enumeration value="parceldropped"/>
     <xs:enumeration value="actiondone"/>
     <xs:enumeration value="navigationdone"/>
    </xs:restriction>
   </xs:simpleType>
  </xs:attribute>
 </xs:complexType>
</xs:element>
```

attribute **msg/pickupConfirmation/@eventType**

type restriction of **xs:string**

properties
- isRef 0
- use optional

facets
- enumeration passengerinvehicle
- enumeration passengerdropped
- enumeration noshow
- enumeration parcelinvehicle
- enumeration parceldropped
- enumeration actiondone
- enumeration navigationdone

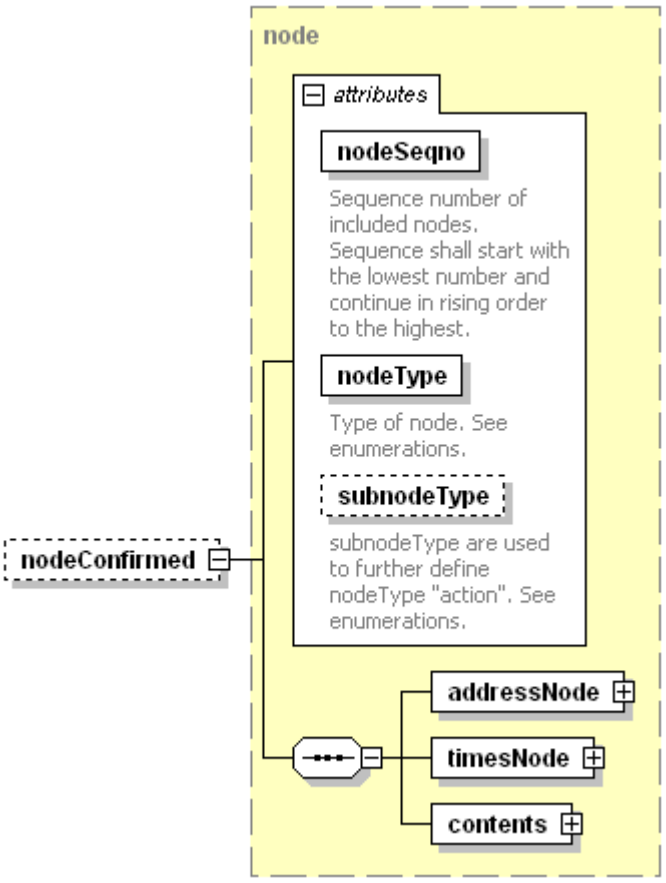
annotation
documentation
Type of event that are beeing confirmed. See enumerations

source

```
<xs:attribute name="eventType" use="optional">
 <xs:annotation>
  <xs:documentation>Type of event that are beeing confirmed. See enumerations</xs:documentation>
 </xs:annotation>
 <xs:simpleType>
  <xs:restriction base="xs:string">
   <xs:enumeration value="passengerinvehicle"/>
   <xs:enumeration value="passengerdropped"/>
   <xs:enumeration value="noshow"/>
   <xs:enumeration value="parcelinvehicle"/>
   <xs:enumeration value="parceldropped"/>
   <xs:enumeration value="actiondone"/>
   <xs:enumeration value="navigationdone"/>
  </xs:restriction>
 </xs:simpleType>
</xs:attribute>
```

element **msg/pickupConfirmation/nodeConfirmed**

diagram



type **node**

properties
isRef 0
minOcc 0
maxOcc 1
content complex

children **addressNode timesNode contents**

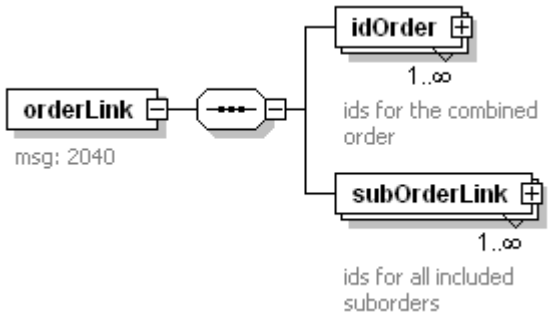
attributes

Name	Type	Use	Default	Fixed	annotation
nodeSeqno	**xs:positiveInteger**	required			documentation Sequence number of included nodes. Sequence shall start with the lowest number and continue in rising order to the highest.
nodeType	**derived by: xs:string**	required			documentation Type of node. See enumerations.
subnodeType	**derived by: xs:string**	optional			documentation subnodeType are used to further define nodeType "action". See enumerations.

source
```
<xs:element name="nodeConfirmed" type="node" minOccurs="0"/>
```

element **msg/orderLink**

diagram



properties
isRef 0
content complex

children **idOrder subOrderLink**

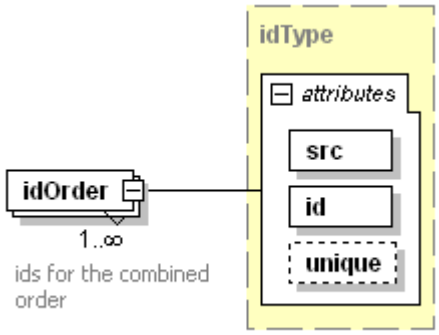
annotation
documentation
msg: 2040

source
```
<xs:element name="orderLink">
 <xs:annotation>
  <xs:documentation>msg: 2040</xs:documentation>
 </xs:annotation>
 <xs:complexType>
  <xs:sequence>
   <xs:element name="idOrder" type="idType" maxOccurs="unbounded">
    <xs:annotation>
     <xs:documentation>ids for the combined order</xs:documentation>
    </xs:annotation>
```

```
    </xs:element>
    <xs:element name="subOrderLink" type="subOrderType" maxOccurs="unbounded">
     <xs:annotation>
      <xs:documentation>ids for all included suborders</xs:documentation>
     </xs:annotation>
    </xs:element>
   </xs:sequence>
  </xs:complexType>
 </xs:element>
```

element **msg/orderLink/idOrder**

diagram



type **idType**

properties
- isRef 0
- minOcc 1
- maxOcc unbounded
- content complex

attributes

Name	Type	Use	Default	Fixed	annotation
src	**xs:string**	required			
id	**xs:string**	required			
unique	**xs:boolean**	optional	false		

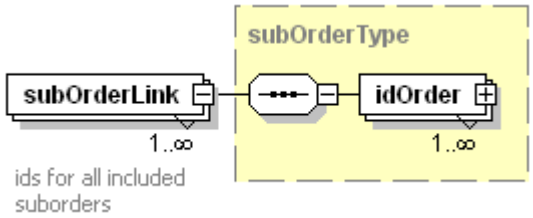
annotation
documentation
ids for the combined order

source

```
<xs:element name="idOrder" type="idType" maxOccurs="unbounded">
 <xs:annotation>
  <xs:documentation>ids for the combined order</xs:documentation>
 </xs:annotation>
</xs:element>
```

element **msg/orderLink/subOrderLink**

diagram



type **subOrderType**

properties
- isRef 0
- minOcc 1
- maxOcc unbounded
- content complex

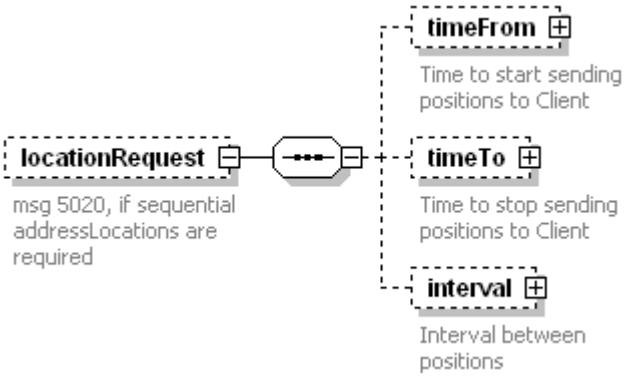
children **idOrder**

annotation
documentation
ids for all included suborders

source
```
<xs:element name="subOrderLink" type="subOrderType" maxOccurs="unbounded">
 <xs:annotation>
  <xs:documentation>ids for all included suborders</xs:documentation>
 </xs:annotation>
</xs:element>
```

element **msg/locationRequest**

diagram



properties

isRef	0
minOcc	0
maxOcc	1
content	complex

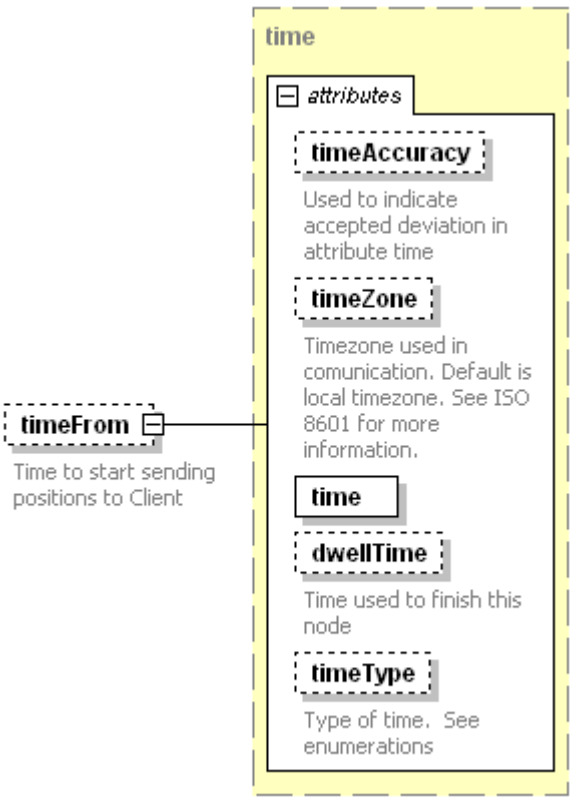
children **timeFrom** **timeTo** **interval**

annotation
documentation
msg 5020, if sequential addressLocations are required

source
```
<xs:element name="locationRequest" minOccurs="0">
 <xs:annotation>
  <xs:documentation>msg 5020, if sequential addressLocations are required</xs:documentation>
 </xs:annotation>
 <xs:complexType>
  <xs:sequence>
   <xs:element name="timeFrom" type="time" minOccurs="0">
    <xs:annotation>
     <xs:documentation>Time to start sending positions to Client</xs:documentation>
    </xs:annotation>
   </xs:element>
   <xs:element name="timeTo" type="time" minOccurs="0">
    <xs:annotation>
     <xs:documentation>Time to stop sending positions to Client</xs:documentation>
    </xs:annotation>
   </xs:element>
   <xs:element name="interval" minOccurs="0">
    <xs:annotation>
     <xs:documentation>Interval between positions</xs:documentation>
    </xs:annotation>
    <xs:complexType>
     <xs:attribute name="seconds" type="xs:integer" use="optional"/>
     <xs:attribute name="meter" type="xs:integer" use="optional"/>
    </xs:complexType>
   </xs:element>
  </xs:sequence>
 </xs:complexType>
</xs:element>
```

element **msg/locationRequest/timeFrom**

diagram



type **time**

properties

isRef	0
minOcc	0
maxOcc	1
content	complex

attributes

Name	Type	Use	Default	Fixed	annotation
timeAccuracy	**xs:string**	optional			documentation Used to indicate accepted deviation in attribute time
timeZone	**xs:integer**	optional			documentation Timezone used in comunication. Default is local timezone. See ISO 8601 for more information.
time	**xs:dateTime**	required			
dwellTime	**xs:int**	optional			documentation Time used to finish this node
timeType	**derived by: xs:string**	optional			documentation

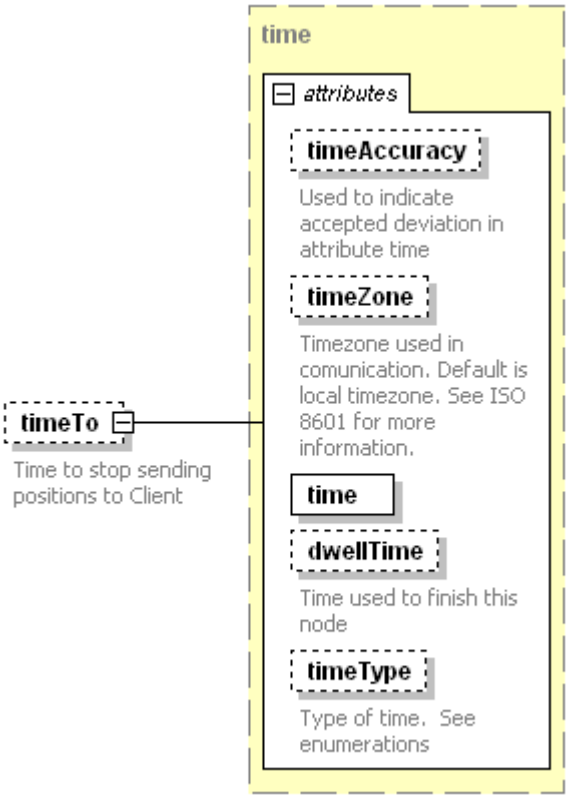
Type of time. See enumerations

annotation documentation
Time to start sending positions to Client

source
```
<xs:element name="timeFrom" type="time" minOccurs="0">
 <xs:annotation>
  <xs:documentation>Time to start sending positions to Client</xs:documentation>
 </xs:annotation>
</xs:element>
```

element **msg/locationRequest/timeTo**

diagram



type **time**

properties
isRef 0
minOcc 0
maxOcc 1
content complex

attributes

Name	Type	Use	Default	Fixed	annotation
timeAccuracy	**xs:string**	optional			documentation Used to indicate accepted deviation in attribute time
timeZone	**xs:integer**	optional			documentation Timezone used in comunication. Default is local timezone. See

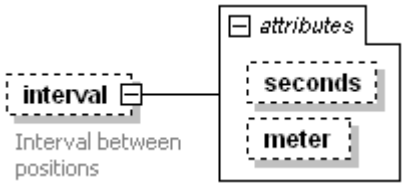
			ISO 8601 for more information.
time	**xs:dateTime**	required	
dwellTime	**xs:int**	optional	documentation Time used to finish this node
timeType	**derived by: xs:string**	optional	documentation Type of time. See enumerations

annotation
documentation
Time to stop sending positions to Client

source
```
<xs:element name="timeTo" type="time" minOccurs="0">
 <xs:annotation>
  <xs:documentation>Time to stop sending positions to Client</xs:documentation>
 </xs:annotation>
</xs:element>
```

element **msg/locationRequest/interval**

diagram



properties
isRef 0
minOcc 0
maxOcc 1
content complex

attributes

Name	Type	Use	Default	Fixed	annotation
seconds	**xs:integer**	optional			
meter	**xs:integer**	optional			

annotation
documentation
Interval between positions

source
```
<xs:element name="interval" minOccurs="0">
 <xs:annotation>
  <xs:documentation>Interval between positions</xs:documentation>
 </xs:annotation>
 <xs:complexType>
  <xs:attribute name="seconds" type="xs:integer" use="optional"/>
  <xs:attribute name="meter" type="xs:integer" use="optional"/>
 </xs:complexType>
</xs:element>
```

attribute **msg/locationRequest/interval/@seconds**

type **xs:integer**

properties
isRef 0
use optional

source
```
<xs:attribute name="seconds" type="xs:integer" use="optional"/>
```

attribute **msg/locationRequest/interval/@meter**

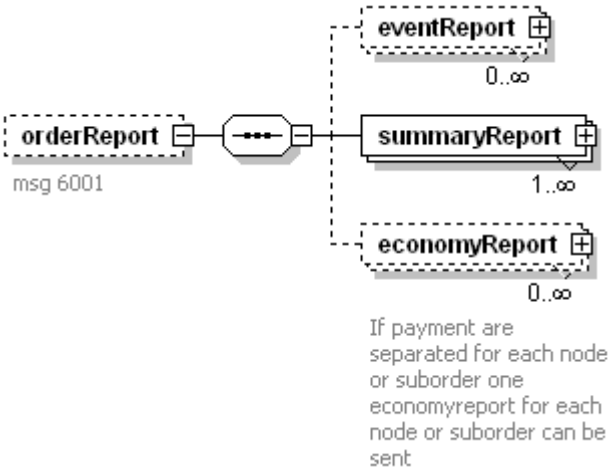
type **xs:integer**

properties
isRef 0
use optional

source
```
<xs:attribute name="meter" type="xs:integer" use="optional"/>
```

element **msg/orderReport**

diagram



properties
isRef 0
minOcc 0
maxOcc 1
content complex

children **eventReport** **summaryReport** **economyReport**

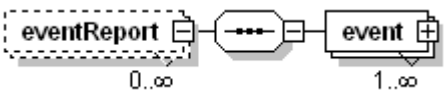
annotation
documentation
msg 6001

source
```
<xs:element name="orderReport" minOccurs="0">
 <xs:annotation>
  <xs:documentation>msg 6001</xs:documentation>
 </xs:annotation>
 <xs:complexType>
  <xs:sequence>
   <xs:element name="eventReport" minOccurs="0" maxOccurs="unbounded">
    <xs:complexType>
     <xs:sequence>
      <xs:element name="event" maxOccurs="unbounded">
       <xs:complexType>
        <xs:sequence>
         <xs:element name="eventTime" type="time" minOccurs="0"/>
         <xs:element name="subOrderEvent" type="subOrderType" minOccurs="0"/>
        </xs:sequence>
        <xs:attribute name="nodeSeqno" type="xs:positiveInteger"/>
        <xs:attribute name="eventType" use="required">
         <xs:simpleType>
          <xs:restriction base="xs:string">
           <xs:enumeration value="acceptOrder"/>
           <xs:enumeration value="start"/>
           <xs:enumeration value="stop"/>
           <xs:enumeration value="pickup"/>
           <xs:enumeration value="destination"/>
           <xs:enumeration value="navigation"/>
           <xs:enumeration value="action"/>
          </xs:restriction>
         </xs:simpleType>
        </xs:attribute>
       </xs:complexType>
      </xs:element>
     </xs:sequence>
    </xs:complexType>
```

```
    </xs:element>
    <xs:element name="summaryReport" maxOccurs="unbounded">
     <xs:complexType>
      <xs:sequence>
       <xs:element name="subOrderSummary" type="subOrderType" minOccurs="0"/>
      </xs:sequence>
      <xs:attribute name="orderEnded" type="xs:boolean" use="required">
       <xs:annotation>
        <xs:documentation>Indicates that this order is finished.</xs:documentation>
       </xs:annotation>
      </xs:attribute>
      <xs:attribute name="distanceStart" type="xs:integer">
       <xs:annotation>
        <xs:documentation>Distance from start of order measured in meters</xs:documentation>
       </xs:annotation>
      </xs:attribute>
      <xs:attribute name="durationStart" type="xs:integer">
       <xs:annotation>
        <xs:documentation>Duration from start of order measured in seconds</xs:documentation>
       </xs:annotation>
      </xs:attribute>
      <xs:attribute name="vehicle" type="xs:string">
       <xs:annotation>
        <xs:documentation>Vehicle that has performed this order</xs:documentation>
       </xs:annotation>
      </xs:attribute>
     </xs:complexType>
    </xs:element>
    <xs:element name="economyReport" minOccurs="0" maxOccurs="unbounded">
     <xs:annotation>
      <xs:documentation>If payment are separated for each node or suborder one economyreport for
each node or suborder can be sent</xs:documentation>
     </xs:annotation>
     <xs:complexType>
      <xs:sequence>
       <xs:element name="payment" type="formOfPayment"/>
       <xs:element name="subOrderEconomy" type="subOrderType" minOccurs="0">
        <xs:annotation>
         <xs:documentation>Used if economyreport are sent for each suborder</xs:documentation>
        </xs:annotation>
       </xs:element>
      </xs:sequence>
      <xs:attribute name="nodeSeqno" type="xs:positiveInteger">
       <xs:annotation>
        <xs:documentation>Used if economyreport are sent for each node</xs:documentation>
       </xs:annotation>
      </xs:attribute>
     </xs:complexType>
    </xs:element>
   </xs:sequence>
  </xs:complexType>
</xs:element>
```

element **msg/orderReport/eventReport**

diagram



properties

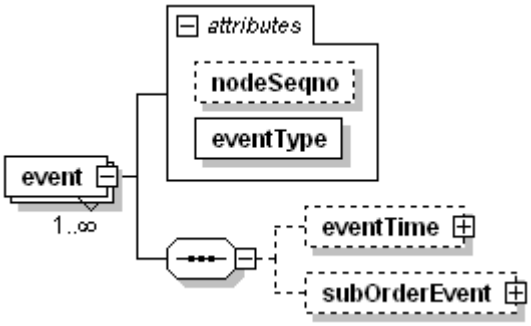
isRef	0
minOcc	0
maxOcc	unbounded
content	complex

children **event**

source
```
<xs:element name="eventReport" minOccurs="0" maxOccurs="unbounded">
 <xs:complexType>
  <xs:sequence>
   <xs:element name="event" maxOccurs="unbounded">
    <xs:complexType>
     <xs:sequence>
      <xs:element name="eventTime" type="time" minOccurs="0"/>
      <xs:element name="subOrderEvent" type="subOrderType" minOccurs="0"/>
     </xs:sequence>
     <xs:attribute name="nodeSeqno" type="xs:positiveInteger"/>
     <xs:attribute name="eventType" use="required">
      <xs:simpleType>
       <xs:restriction base="xs:string">
        <xs:enumeration value="acceptOrder"/>
        <xs:enumeration value="start"/>
        <xs:enumeration value="stop"/>
        <xs:enumeration value="pickup"/>
        <xs:enumeration value="destination"/>
        <xs:enumeration value="navigation"/>
        <xs:enumeration value="action"/>
       </xs:restriction>
      </xs:simpleType>
     </xs:attribute>
    </xs:complexType>
   </xs:element>
  </xs:sequence>
 </xs:complexType>
</xs:element>
```

element **msg/orderReport/eventReport/event**

diagram



properties

isRef	0
minOcc	1
maxOcc	unbounded
content	complex

children **eventTime** **subOrderEvent**

attributes

Name	Type	Use	Default	Fixed	annotation
nodeSeqno	**xs:positiveInteger**				
eventType	**derived by: xs:string**	required			

source
```
<xs:element name="event" maxOccurs="unbounded">
 <xs:complexType>
  <xs:sequence>
   <xs:element name="eventTime" type="time" minOccurs="0"/>
   <xs:element name="subOrderEvent" type="subOrderType" minOccurs="0"/>
  </xs:sequence>
  <xs:attribute name="nodeSeqno" type="xs:positiveInteger"/>
  <xs:attribute name="eventType" use="required">
   <xs:simpleType>
    <xs:restriction base="xs:string">
     <xs:enumeration value="acceptOrder"/>
```

```
            <xs:enumeration value="start"/>
            <xs:enumeration value="stop"/>
            <xs:enumeration value="pickup"/>
            <xs:enumeration value="destination"/>
            <xs:enumeration value="navigation"/>
            <xs:enumeration value="action"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:attribute>
    </xs:complexType>
  </xs:element>
```

attribute **msg/orderReport/eventReport/event/@nodeSeqno**

type **xs:positiveInteger**

properties
- isRef 0

source

```
<xs:attribute name="nodeSeqno" type="xs:positiveInteger"/>
```

attribute **msg/orderReport/eventReport/event/@eventType**

type restriction of **xs:string**

properties
- isRef 0
- use required

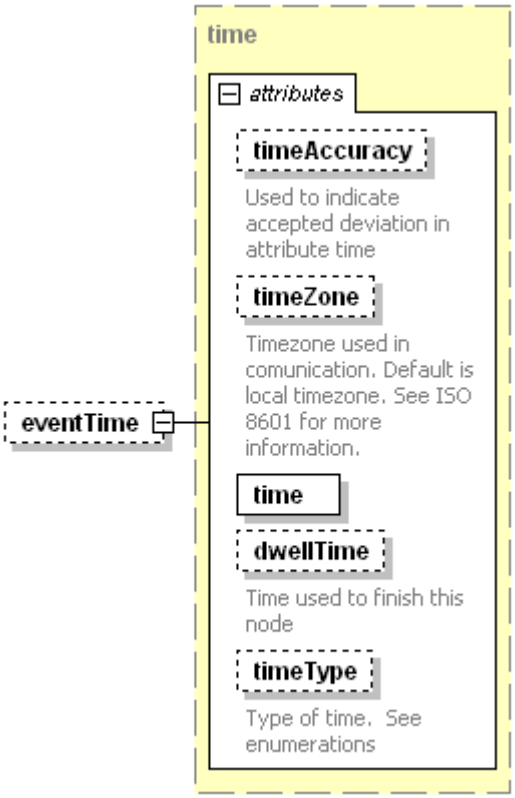
facets
- enumeration acceptOrder
- enumeration start
- enumeration stop
- enumeration pickup
- enumeration destination
- enumeration navigation
- enumeration action

source

```
<xs:attribute name="eventType" use="required">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="acceptOrder"/>
      <xs:enumeration value="start"/>
      <xs:enumeration value="stop"/>
      <xs:enumeration value="pickup"/>
      <xs:enumeration value="destination"/>
      <xs:enumeration value="navigation"/>
      <xs:enumeration value="action"/>
    </xs:restriction>
  </xs:simpleType>
</xs:attribute>
```

element **msg/orderReport/eventReport/event/eventTime**

diagram



type **time**

properties

isRef	0
minOcc	0
maxOcc	1
content	complex

attributes

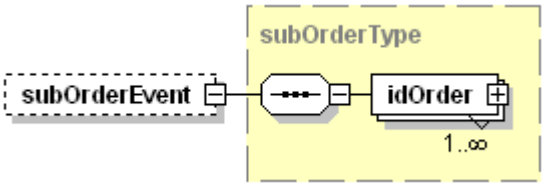
Name	Type	Use	Default	Fixed	annotation
timeAccuracy	**xs:string**	optional			documentation Used to indicate accepted deviation in attribute time
timeZone	**xs:integer**	optional			documentation Timezone used in comunication. Default is local timezone. See ISO 8601 for more information.
time	**xs:dateTime**	required			
dwellTime	**xs:int**	optional			documentation Time used to finish this node
timeType	**derived by: xs:string**	optional			documentation

Type of time. See enumerations

source <xs:element name="eventTime" type="time" minOccurs="0"/>

element **msg/orderReport/eventReport/event/subOrderEvent**

diagram



type **subOrderType**

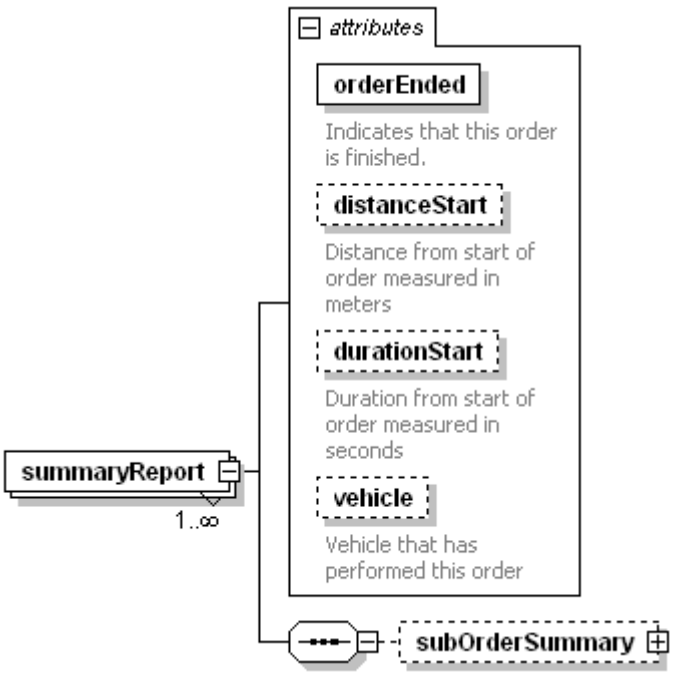
properties
- isRef 0
- minOcc 0
- maxOcc 1
- content complex

children **idOrder**

source <xs:element name="subOrderEvent" type="subOrderType" minOccurs="0"/>

element **msg/orderReport/summaryReport**

diagram



properties
- isRef 0
- minOcc 1
- maxOcc unbounded
- content complex

children **subOrderSummary**

attributes

Name	Type	Use	Default	Fixed	annotation
orderEnded	**xs:boolean**	required			documentation Indicates that this order is finished.
distanceStart	**xs:integer**				documentatio

		n Distance from start of order measured in meters
durationStart	**xs:integer**	documentation Duration from start of order measured in seconds
vehicle	**xs:string**	documentation Vehicle that has performed this order

source
```
<xs:element name="summaryReport" maxOccurs="unbounded">
 <xs:complexType>
  <xs:sequence>
   <xs:element name="subOrderSummary" type="subOrderType" minOccurs="0"/>
  </xs:sequence>
  <xs:attribute name="orderEnded" type="xs:boolean" use="required">
   <xs:annotation>
    <xs:documentation>Indicates that this order is finished.</xs:documentation>
   </xs:annotation>
  </xs:attribute>
  <xs:attribute name="distanceStart" type="xs:integer">
   <xs:annotation>
    <xs:documentation>Distance from start of order measured in meters</xs:documentation>
   </xs:annotation>
  </xs:attribute>
  <xs:attribute name="durationStart" type="xs:integer">
   <xs:annotation>
    <xs:documentation>Duration from start of order measured in seconds</xs:documentation>
   </xs:annotation>
  </xs:attribute>
  <xs:attribute name="vehicle" type="xs:string">
   <xs:annotation>
    <xs:documentation>Vehicle that has performed this order</xs:documentation>
   </xs:annotation>
  </xs:attribute>
 </xs:complexType>
</xs:element>
```

attribute **msg/orderReport/summaryReport/@orderEnded**

type **xs:boolean**

properties
- isRef 0
- use required

annotation
documentation
Indicates that this order is finished.

source
```
<xs:attribute name="orderEnded" type="xs:boolean" use="required">
 <xs:annotation>
  <xs:documentation>Indicates that this order is finished.</xs:documentation>
 </xs:annotation>
</xs:attribute>
```

attribute **msg/orderReport/summaryReport/@distanceStart**

type **xs:integer**

properties
- isRef 0

annotation documentation
Distance from start of order measured in meters

source
```
<xs:attribute name="distanceStart" type="xs:integer">
 <xs:annotation>
  <xs:documentation>Distance from start of order measured in meters</xs:documentation>
 </xs:annotation>
</xs:attribute>
```

attribute **msg/orderReport/summaryReport/@durationStart**

type **xs:integer**

properties isRef 0

annotation documentation
Duration from start of order measured in seconds

source
```
<xs:attribute name="durationStart" type="xs:integer">
 <xs:annotation>
  <xs:documentation>Duration from start of order measured in seconds</xs:documentation>
 </xs:annotation>
</xs:attribute>
```

attribute **msg/orderReport/summaryReport/@vehicle**

type **xs:string**

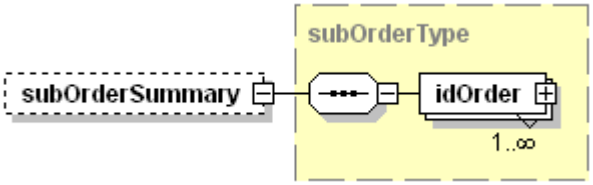
properties isRef 0

annotation documentation
Vehicle that has performed this order

source
```
<xs:attribute name="vehicle" type="xs:string">
 <xs:annotation>
  <xs:documentation>Vehicle that has performed this order</xs:documentation>
 </xs:annotation>
</xs:attribute>
```

element **msg/orderReport/summaryReport/subOrderSummary**

diagram



type **subOrderType**

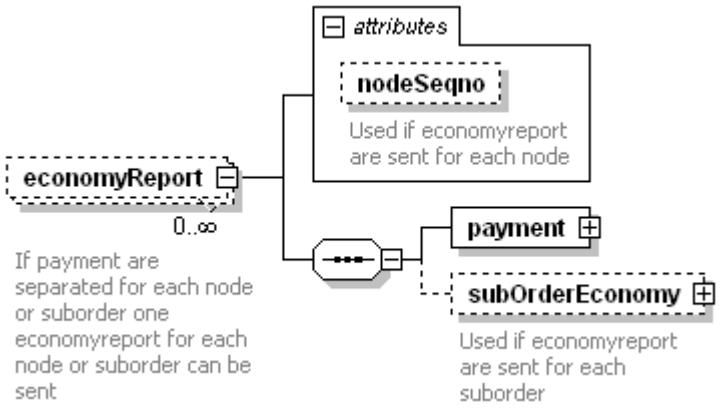
properties
isRef 0
minOcc 0
maxOcc 1
content complex

children **idOrder**

source
```
<xs:element name="subOrderSummary" type="subOrderType" minOccurs="0"/>
```

element **msg/orderReport/economyReport**

diagram



properties

isRef	0
minOcc	0
maxOcc	unbounded
content	complex

children **payment** **subOrderEconomy**

attributes

Name	Type	Use	Default	Fixed	annotation
nodeSeqno	**xs:positiveInteger**				documentation Used if economyreport are sent for each node

annotation
documentation
If payment are separated for each node or suborder one economyreport for each node or suborder can be sent

source

```
<xs:element name="economyReport" minOccurs="0" maxOccurs="unbounded">
 <xs:annotation>
  <xs:documentation>If payment are separated for each node or suborder one economyreport for each node or suborder can be sent</xs:documentation>
 </xs:annotation>
 <xs:complexType>
  <xs:sequence>
   <xs:element name="payment" type="formOfPayment"/>
   <xs:element name="subOrderEconomy" type="subOrderType" minOccurs="0">
    <xs:annotation>
     <xs:documentation>Used if economyreport are sent for each suborder</xs:documentation>
    </xs:annotation>
   </xs:element>
  </xs:sequence>
  <xs:attribute name="nodeSeqno" type="xs:positiveInteger">
   <xs:annotation>
    <xs:documentation>Used if economyreport are sent for each node</xs:documentation>
   </xs:annotation>
  </xs:attribute>
 </xs:complexType>
</xs:element>
```

attribute **msg/orderReport/economyReport/@nodeSeqno**

type **xs:positiveInteger**

properties

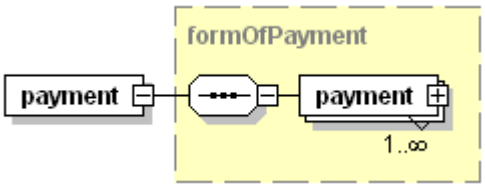
isRef	0

annotation
documentation
Used if economyreport are sent for each node

source
```
<xs:attribute name="nodeSeqno" type="xs:positiveInteger">
  <xs:annotation>
    <xs:documentation>Used if economyreport are sent for each node</xs:documentation>
  </xs:annotation>
</xs:attribute>
```

element **msg/orderReport/economyReport/payment**

diagram



type **formOfPayment**

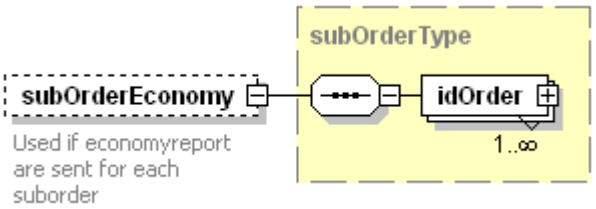
properties
- isRef 0
- content complex

children **payment**

source
```
<xs:element name="payment" type="formOfPayment"/>
```

element **msg/orderReport/economyReport/subOrderEconomy**

diagram



type **subOrderType**

properties
- isRef 0
- minOcc 0
- maxOcc 1
- content complex

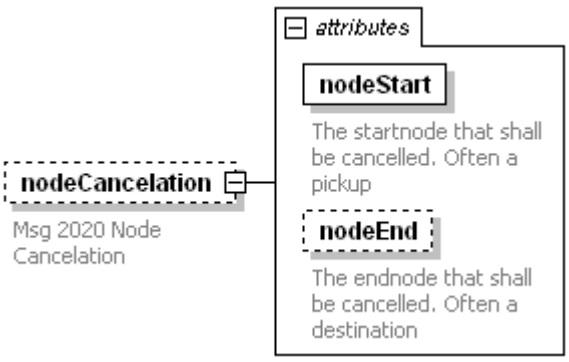
children **idOrder**

annotation
documentation
Used if economyreport are sent for each suborder

source
```
<xs:element name="subOrderEconomy" type="subOrderType" minOccurs="0">
  <xs:annotation>
    <xs:documentation>Used if economyreport are sent for each suborder</xs:documentation>
  </xs:annotation>
</xs:element>
```

element **msg/nodeCancelation**

diagram



properties

isRef	0
minOcc	0
maxOcc	1
content	complex

attributes

Name	Type	Use	Default	Fixed	annotation
nodeStart	**xs:positiveInteger**	required			documentation The startnode that shall be cancelled. Often a pickup
nodeEnd	**xs:positiveInteger**				documentation The endnode that shall be cancelled. Often a destination

annotation

documentation
Msg 2020 Node Cancelation

source

```
<xs:element name="nodeCancelation" minOccurs="0">
 <xs:annotation>
  <xs:documentation>Msg 2020 Node Cancelation</xs:documentation>
 </xs:annotation>
 <xs:complexType>
  <xs:attribute name="nodeStart" type="xs:positiveInteger" use="required">
   <xs:annotation>
    <xs:documentation>The startnode that shall be cancelled. Often a pickup</xs:documentation>
   </xs:annotation>
  </xs:attribute>
  <xs:attribute name="nodeEnd" type="xs:positiveInteger">
   <xs:annotation>
    <xs:documentation>The endnode that shall be cancelled. Often a destination</xs:documentation>
   </xs:annotation>
  </xs:attribute>
 </xs:complexType>
</xs:element>
```

attribute **msg/nodeCancelation/@nodeStart**

type **xs:positiveInteger**

properties

isRef	0
use	required

annotation

documentation
The startnode that shall be cancelled. Often a pickup

source

```
<xs:attribute name="nodeStart" type="xs:positiveInteger" use="required">
```

```
  <xs:annotation>
    <xs:documentation>The startnode that shall be cancelled. Often a pickup</xs:documentation>
  </xs:annotation>
</xs:attribute>
```

attribute **msg/nodeCancelation/@nodeEnd**

type **xs:positiveInteger**

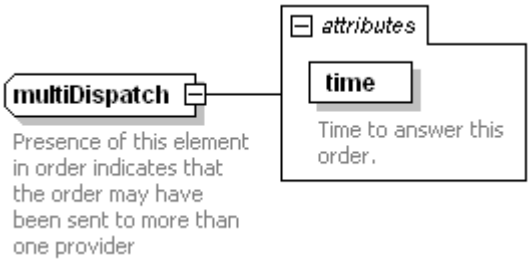
properties isRef 0

annotation documentation
The endnode that shall be cancelled. Often a destination

source
```
<xs:attribute name="nodeEnd" type="xs:positiveInteger">
  <xs:annotation>
    <xs:documentation>The endnode that shall be cancelled. Often a destination</xs:documentation>
  </xs:annotation>
</xs:attribute>
```

complexType **multiDispatch**

diagram



used by element **process/multiDispatch**

attributes

Name	Type	Use	Default	Fixed	annotation
time	**xs:dateTime**	required			documentation Time to answer this order.

annotation documentation
Presence of this element in order indicates that the order may have been sent to more than one provider

source
```
<xs:complexType name="multiDispatch">
  <xs:annotation>
    <xs:documentation>Presence of this element in order indicates that the order may have been sent to
more than one provider</xs:documentation>
  </xs:annotation>
  <xs:attribute name="time" type="xs:dateTime" use="required">
    <xs:annotation>
      <xs:documentation>Time to answer this order.</xs:documentation>
    </xs:annotation>
  </xs:attribute>
</xs:complexType>
```

attribute **multiDispatch/@time**

type **xs:dateTime**

properties isRef 0
use required

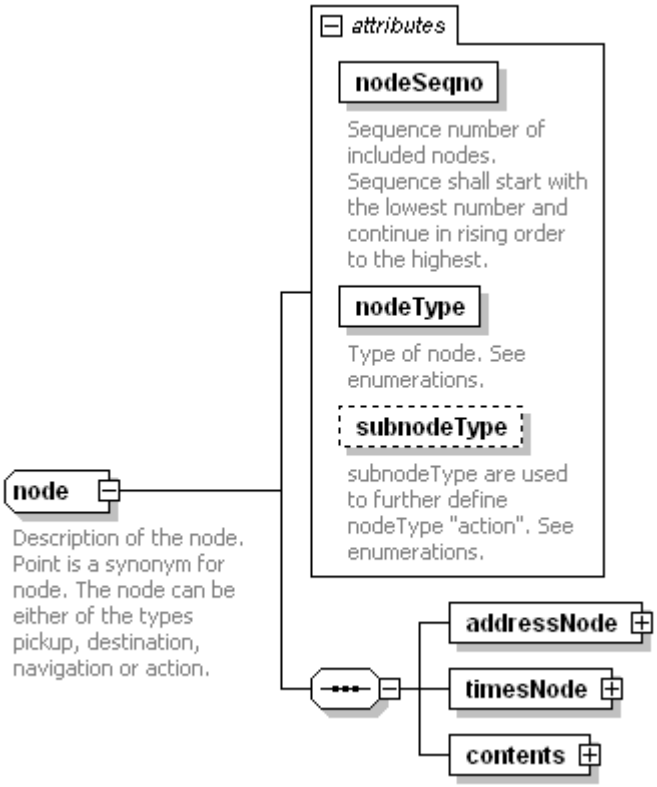
annotation documentation
Time to answer this order.

source
```
<xs:attribute name="time" type="xs:dateTime" use="required">
  <xs:annotation>
    <xs:documentation>Time to answer this order.</xs:documentation>
```

```
    </xs:annotation>
  </xs:attribute>
```

complexType **node**

diagram



children **addressNode** **timesNode** **contents**

used by elements **route/node** **pickupConfirmation/node** **msg/pickupConfirmation/nodeConfirmed**

attributes

Name	Type	Use	Default	Fixed	annotation
nodeSeqno	**xs:positiveInteger**	required			documentation Sequence number of included nodes. Sequence shall start with the lowest number and continue in rising order to the highest.
nodeType	**derived by: xs:string**	required			documentation Type of node. See enumerations.
subnodeType	**derived by: xs:string**	optional			documentation subnodeType are used to further define nodeType "action". See enumerations.

annotation documentation
Description of the node. Point is a synonym for node. The node can be either of the types pickup, destination, navigation or action.

source
```
<xs:complexType name="node">
 <xs:annotation>
  <xs:documentation>Description of the node. Point is a synonym for node. The node can be either of the types pickup, destination, navigation or action.</xs:documentation>
 </xs:annotation>
 <xs:sequence>
  <xs:element name="addressNode">
   <xs:complexType>
    <xs:complexContent>
     <xs:extension base="addressType"/>
    </xs:complexContent>
   </xs:complexType>
  </xs:element>
  <xs:element name="timesNode" type="timesType"/>
  <xs:element name="contents" type="contents"/>
 </xs:sequence>
 <xs:attribute name="nodeSeqno" type="xs:positiveInteger" use="required">
  <xs:annotation>
   <xs:documentation>Sequence number of included nodes. Sequence shall start with the lowest number and continue in rising order to the highest. </xs:documentation>
  </xs:annotation>
 </xs:attribute>
 <xs:attribute name="nodeType" use="required">
  <xs:annotation>
   <xs:documentation>Type of node. See enumerations.</xs:documentation>
  </xs:annotation>
  <xs:simpleType>
   <xs:restriction base="xs:string"/>
  </xs:simpleType>
 </xs:attribute>
 <xs:attribute name="subnodeType" use="optional">
  <xs:annotation>
   <xs:documentation>subnodeType are used to further define nodeType "action". See enumerations.</xs:documentation>
  </xs:annotation>
  <xs:simpleType>
   <xs:restriction base="xs:string"/>
  </xs:simpleType>
 </xs:attribute>
</xs:complexType>
```

attribute **node/@nodeSeqno**

type **xs:positiveInteger**

properties
isRef 0
use required

annotation documentation
Sequence number of included nodes. Sequence shall start with the lowest number and continue in rising order to the highest.

source
```
<xs:attribute name="nodeSeqno" type="xs:positiveInteger" use="required">
 <xs:annotation>
  <xs:documentation>Sequence number of included nodes. Sequence shall start with the lowest number and continue in rising order to the highest. </xs:documentation>
 </xs:annotation>
</xs:attribute>
```

attribute **node/@nodeType**

type restriction of **xs:string**

properties
  isRef 0
  use required

annotation
  documentation
  Type of node. See enumerations.

source
```
<xs:attribute name="nodeType" use="required">
 <xs:annotation>
  <xs:documentation>Type of node. See enumerations.</xs:documentation>
 </xs:annotation>
 <xs:simpleType>
  <xs:restriction base="xs:string"/>
 </xs:simpleType>
</xs:attribute>
```

attribute **node/@subnodeType**

type restriction of **xs:string**

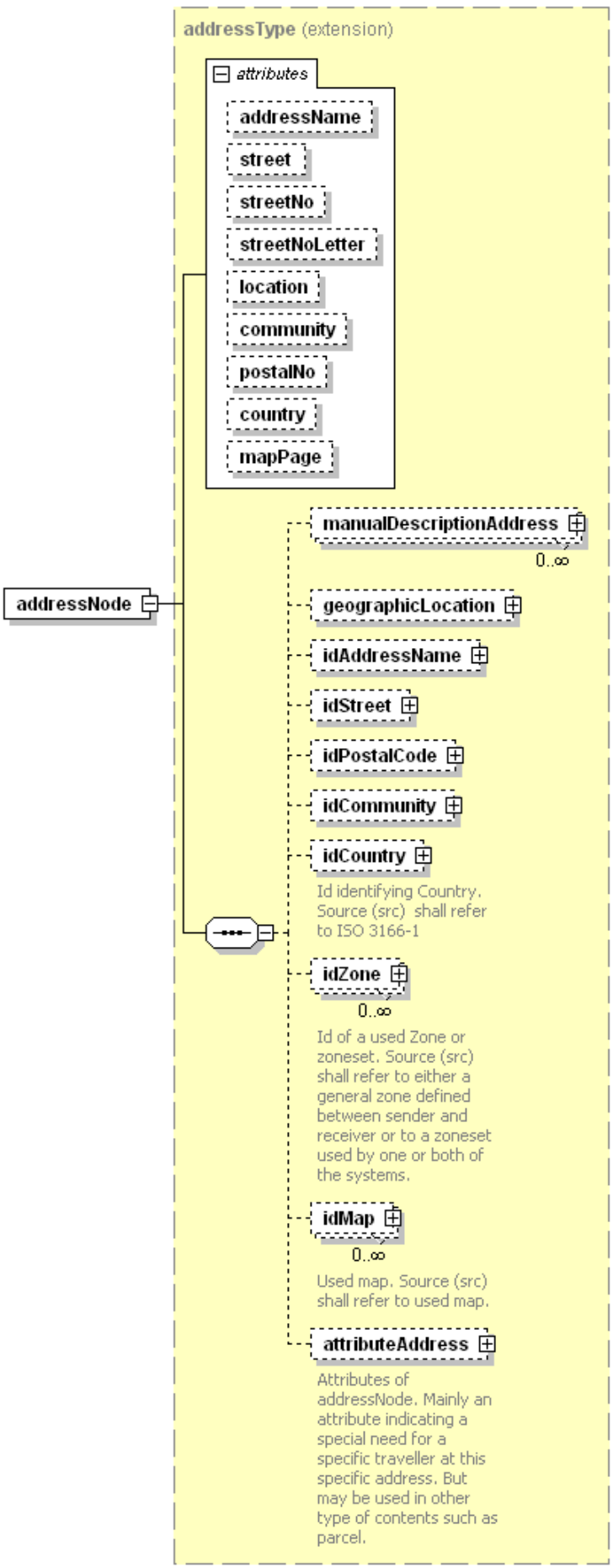
properties
  isRef 0
  use optional

annotation
  documentation
  subnodeType are used to further define nodeType "action". See enumerations.

source
```
<xs:attribute name="subnodeType" use="optional">
 <xs:annotation>
  <xs:documentation>subnodeType are used to further define nodeType "action". See
enumerations.</xs:documentation>
 </xs:annotation>
 <xs:simpleType>
  <xs:restriction base="xs:string"/>
 </xs:simpleType>
</xs:attribute>
```

element **node/addressNode**

diagram


addressType (extension)
attributes
addressName
street
streetNo
streetNoLetter
location
community
postalNo
country
mapPage
addressNode
manualDescriptionAddress
0..∞
geographicLocation
idAddressName
idStreet
idPostalCode
idCommunity
idCountry
Id identifying Country. Source (src) shall refer to ISO 3166-1
idZone
0..∞
Id of a used Zone or zoneset. Source (src) shall refer to either a general zone defined between sender and receiver or to a zoneset used by one or both of the systems.
idMap
0..∞
Used map. Source (src) shall refer to used map.
attributeAddress
Attributes of addressNode. Mainly an attribute indicating a special need for a specific traveller at this specific address. But may be used in other type of contents such as parcel.

type extension of **addressType**

properties
isRef 0
content complex

children **manualDescriptionAddress** **geographicLocation** **idAddressName** **idStreet** **idPostalCode** **idCommunity** **idCountry** **idZone** **idMap** **attributeAddress**

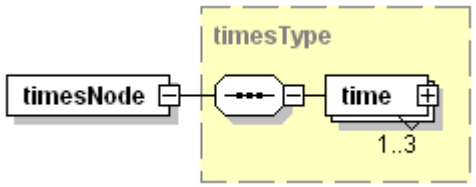
attributes

Name	Type	Use	Default	Fixed	annotation
addressName	**xs:string**	optional			
street	**xs:string**	optional			
streetNo	**xs:positiveInteger**	optional			
streetNoLetter	**xs:string**	optional			
location	**xs:string**	optional			
community	**xs:string**	optional			
postalNo	**xs:string**	optional			
country	**xs:string**	optional			
mapPage	**xs:string**	optional			

source
```
<xs:element name="addressNode">
 <xs:complexType>
  <xs:complexContent>
   <xs:extension base="addressType"/>
  </xs:complexContent>
 </xs:complexType>
</xs:element>
```

element **node/timesNode**

diagram



type **timesType**

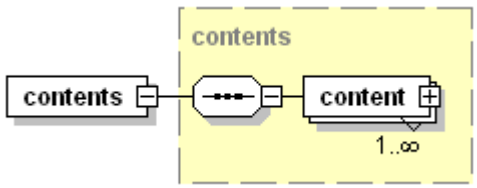
properties
isRef 0
content complex

children **time**

source `<xs:element name="timesNode" type="timesType"/>`

element **node/contents**

diagram



type **contents**

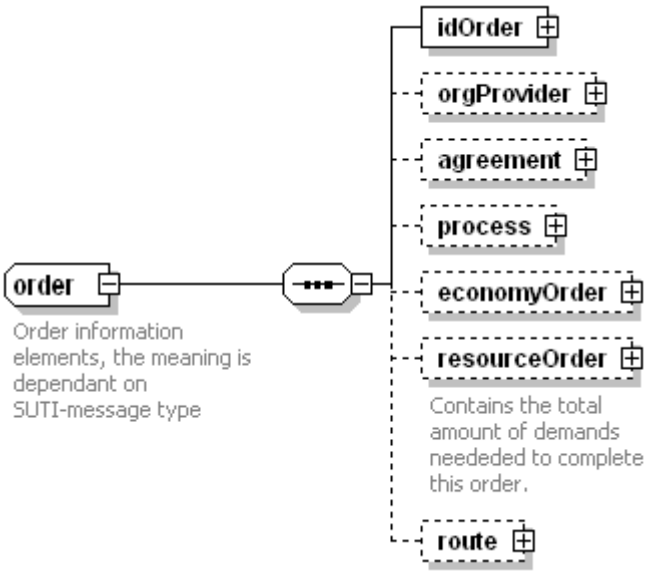
properties
isRef 0
content complex

children **content**

source `<xs:element name="contents" type="contents"/>`

complexType **order**

diagram



children **idOrder** **orgProvider** **agreement** **process** **economyOrder** **resourceOrder** **route**

used by element **msg/order**

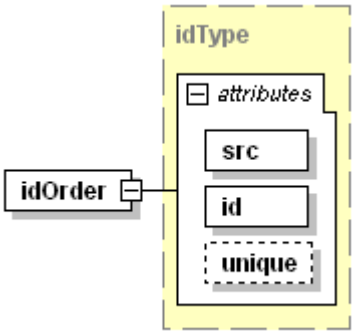
annotation documentation
Order information elements, the meaning is dependant on SUTI-message type

source

```
<xs:complexType name="order">
 <xs:annotation>
  <xs:documentation>Order information elements, the meaning is dependant on SUTI-message type</xs:documentation>
 </xs:annotation>
 <xs:sequence>
  <xs:element name="idOrder" type="idType"/>
  <xs:element name="orgProvider" type="orgType" minOccurs="0"/>
  <xs:element name="agreement" type="agreement" minOccurs="0"/>
  <xs:element name="process" minOccurs="0">
   <xs:complexType>
    <xs:complexContent>
     <xs:extension base="process"/>
    </xs:complexContent>
   </xs:complexType>
  </xs:element>
  <xs:element name="economyOrder" type="economyType" minOccurs="0"/>
  <xs:element name="resourceOrder" type="resourceType" minOccurs="0">
   <xs:annotation>
    <xs:documentation>Contains the total amount of demands neededed to complete this order.</xs:documentation>
   </xs:annotation>
  </xs:element>
  <xs:element name="route" type="route" minOccurs="0"/>
 </xs:sequence>
</xs:complexType>
```

element **order/idOrder**

diagram



type **idType**

properties
isRef 0
content complex

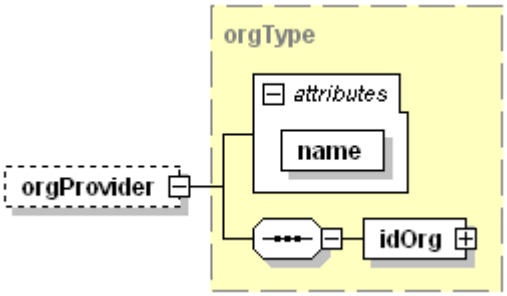
attributes

Name	Type	Use	Default	Fixed	annotation
src	**xs:string**	required			
id	**xs:string**	required			
unique	**xs:boolean**	optional	false		

source
```
<xs:element name="idOrder" type="idType"/>
```

element **order/orgProvider**

diagram



type **orgType**

properties
isRef 0
minOcc 0
maxOcc 1
content complex

children **idOrg**

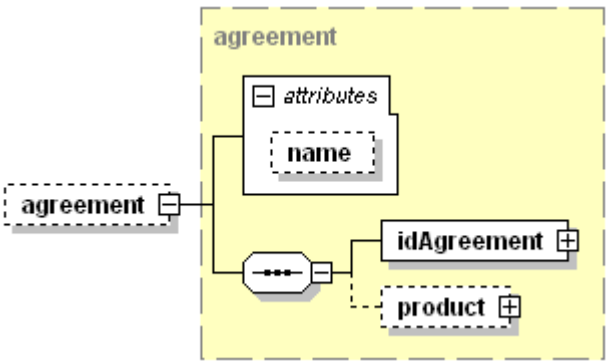
attributes

Name	Type	Use	Default	Fixed	annotation
name	**xs:string**	required			

source
```
<xs:element name="orgProvider" type="orgType" minOccurs="0"/>
```

element **order/agreement**

diagram



type **agreement**

properties

isRef	0
minOcc	0
maxOcc	1
content	complex

children **idAgreement** **product**

attributes

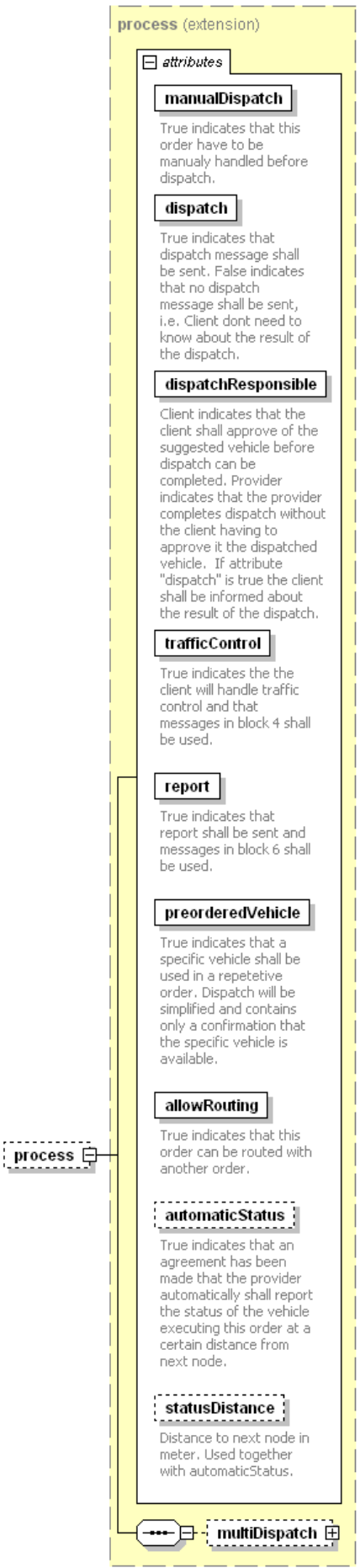
Name	Type	Use	Default	Fixed	annotation
name	**xs:string**	optional			

source

```
<xs:element name="agreement" type="agreement" minOccurs="0"/>
```

element **order/process**

diagram


process (extension)
attributes
manualDispatch
True indicates that this order have to be manualy handled before dispatch.
dispatch
True indicates that dispatch message shall be sent. False indicates that no dispatch message shall be sent, i.e. Client dont need to know about the result of the dispatch.
dispatchResponsible
Client indicates that the client shall approve of the suggested vehicle before dispatch can be completed. Provider indicates that the provider completes dispatch without the client having to approve it the dispatched vehicle. If attribute "dispatch" is true the client shall be informed about the result of the dispatch.
trafficControl
True indicates the the client will handle traffic control and that messages in block 4 shall be used.
report
True indicates that report shall be sent and messages in block 6 shall be used.
preorderedVehicle
True indicates that a specific vehicle shall be used in a repetetive order. Dispatch will be simplified and contains only a confirmation that the specific vehicle is available.
allowRouting
True indicates that this order can be routed with another order.
automaticStatus
True indicates that an agreement has been made that the provider automatically shall report the status of the vehicle executing this order at a certain distance from next node.
statusDistance
Distance to next node in meter. Used together with automaticStatus.
multiDispatch
process

type extension of **process**

properties
- isRef 0
- minOcc 0
- maxOcc 1
- content complex

children **multiDispatch**

attributes

Name	Type	Use	Default	Fixed	annotation
manualDispatch	**xs:boolean**	required			documentation True indicates that this order have to be manualy handled before dispatch.
dispatch	**xs:boolean**	required			documentation True indicates that dispatch message shall be sent. False indicates that no dispatch message shall be sent, i.e. Client dont need to know about the result of the dispatch.
dispatchResponsible	**derived by: xs:string**	required			documentation Client indicates that the client shall approve of the suggested vehicle before dispatch can be completed. Provider indicates that the provider completes dispatch without the client having to approve it the dispatched vehicle. If attribute "dispatch" is true the client shall be informed about the result of the dispatch.
trafficControl	**xs:boolean**	required			documentation True indicates the the client will handle traffic control

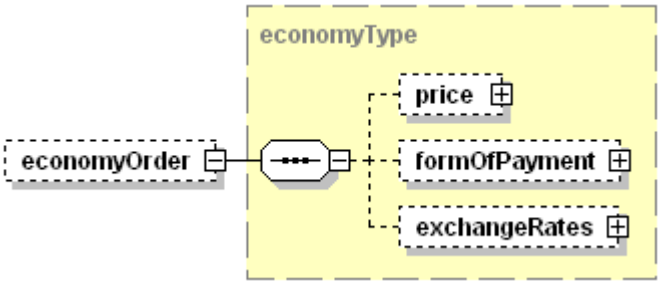
			and that messages in block 4 shall be used.
report	**xs:boolean**	required	documentation True indicates that report shall be sent and messages in block 6 shall be used.
preorderedVehicle	**xs:boolean**	required	documentation True indicates that a specific vehicle shall be used in a repetetive order. Dispatch will be simplified and contains only a confirmation that the specific vehicle is available.
allowRouting	**xs:boolean**	required	documentation True indicates that this order can be routed with another order.
automaticStatus	**xs:boolean**	optional	documentation True indicates that an agreement has been made that the provider automatically shall report the status of the vehicle executing this order at a certain distance from next node.
statusDistance	**xs:nonNegativeInteger**	optional	documentation Distance to next node in meter. Used together with automaticStat

us.

source
```
<xs:element name="process" minOccurs="0">
  <xs:complexType>
    <xs:complexContent>
      <xs:extension base="process"/>
    </xs:complexContent>
  </xs:complexType>
</xs:element>
```

element **order/economyOrder**

diagram



type **economyType**

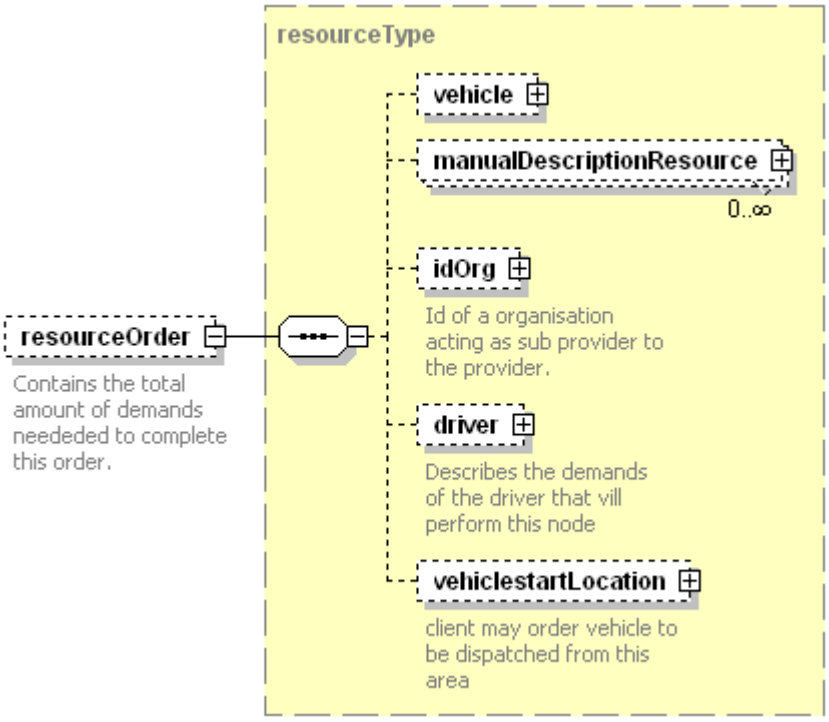
properties
- isRef 0
- minOcc 0
- maxOcc 1
- content complex

children **price** **formOfPayment** **exchangeRates**

source
```
<xs:element name="economyOrder" type="economyType" minOccurs="0"/>
```

element **order/resourceOrder**

diagram



type **resourceType**

properties
isRef 0
minOcc 0
maxOcc 1
content complex

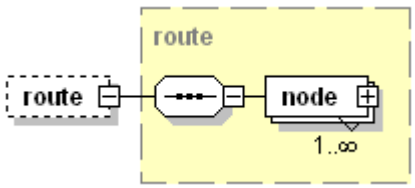
children **vehicle** **manualDescriptionResource** **idOrg** **driver** **vehiclestartLocation**

annotation
documentation
Contains the total amount of demands neededed to complete this order.

source
```
<xs:element name="resourceOrder" type="resourceType" minOccurs="0">
 <xs:annotation>
  <xs:documentation>Contains the total amount of demands neededed to complete this order.</xs:documentation>
 </xs:annotation>
</xs:element>
```

element **order/route**

diagram



type **route**

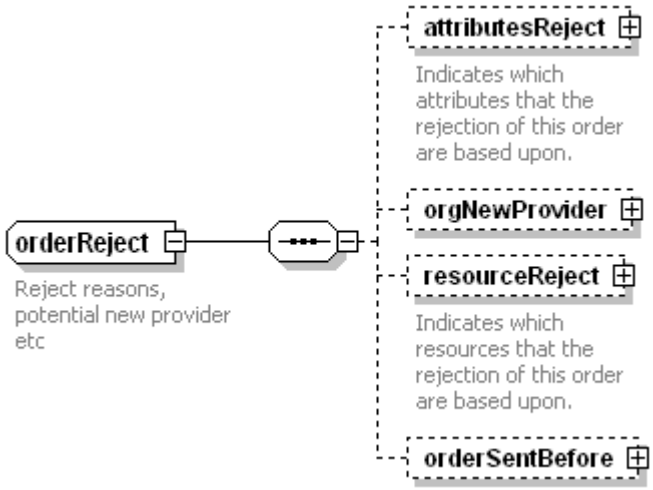
properties
isRef 0
minOcc 0
maxOcc 1
content complex

children **node**

source
```
<xs:element name="route" type="route" minOccurs="0"/>
```

complexType **orderReject**

diagram



children **attributesReject** **orgNewProvider** **resourceReject** **orderSentBefore**

used by
element **msg/orderReject**

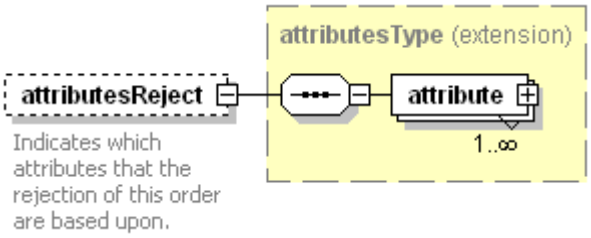
annotation
documentation
Reject reasons, potential new provider etc

source
```
<xs:complexType name="orderReject">
 <xs:annotation>
  <xs:documentation>Reject reasons, potential new provider etc</xs:documentation>
 </xs:annotation>
```

```
 <xs:sequence>
  <xs:element name="attributesReject" minOccurs="0">
   <xs:annotation>
    <xs:documentation>Indicates which attributes that the rejection of this order are based
upon.</xs:documentation>
   </xs:annotation>
   <xs:complexType>
    <xs:complexContent>
     <xs:extension base="attributesType"/>
    </xs:complexContent>
   </xs:complexType>
  </xs:element>
  <xs:element name="orgNewProvider" type="orgType" minOccurs="0"/>
  <xs:element name="resourceReject" type="resourceType" minOccurs="0">
   <xs:annotation>
    <xs:documentation>Indicates which resources that the rejection of this order are based
upon.</xs:documentation>
   </xs:annotation>
  </xs:element>
  <xs:element name="orderSentBefore" minOccurs="0">
   <xs:complexType>
    <xs:sequence>
     <xs:element name="idMsg" type="idType"/>
    </xs:sequence>
   </xs:complexType>
  </xs:element>
 </xs:sequence>
</xs:complexType>
```

element **orderReject/attributesReject**

diagram



type extension of **attributesType**

properties

isRef	0
minOcc	0
maxOcc	1
content	complex

children **attribute**

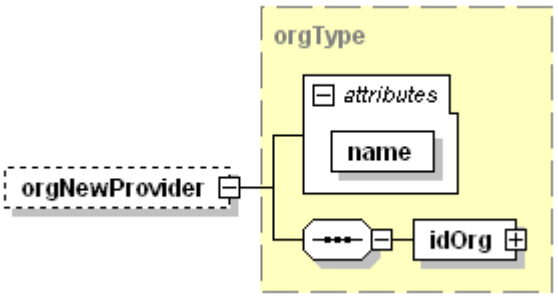
annotation
documentation
Indicates which attributes that the rejection of this order are based upon.

source

```
<xs:element name="attributesReject" minOccurs="0">
 <xs:annotation>
  <xs:documentation>Indicates which attributes that the rejection of this order are based
upon.</xs:documentation>
 </xs:annotation>
 <xs:complexType>
  <xs:complexContent>
   <xs:extension base="attributesType"/>
  </xs:complexContent>
 </xs:complexType>
</xs:element>
```

element **orderReject/orgNewProvider**

diagram



type **orgType**

properties
isRef 0
minOcc 0
maxOcc 1
content complex

children **idOrg**

attributes

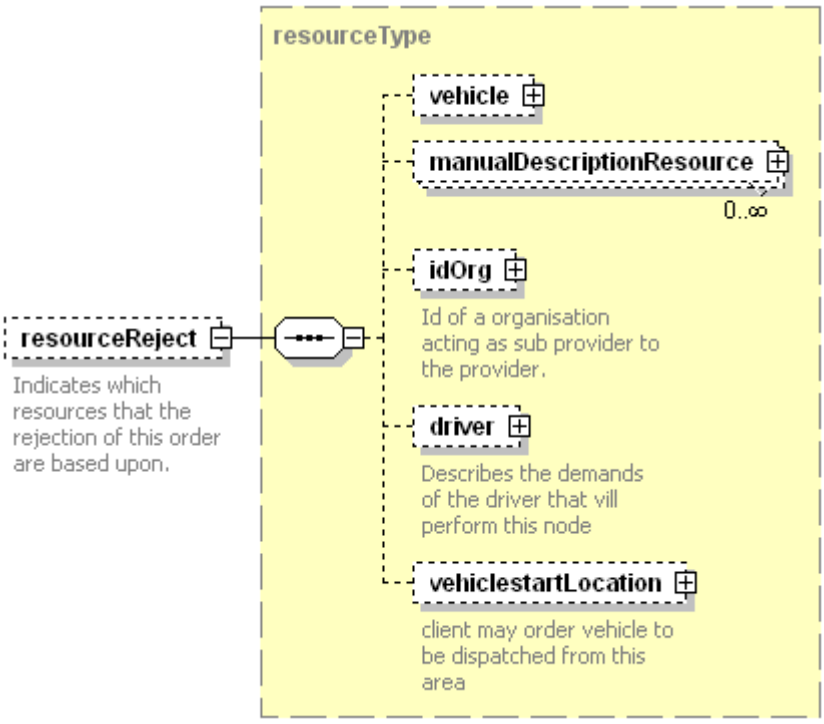
Name	Type	Use	Default	Fixed	annotation
name	**xs:string**	required			

source

```
<xs:element name="orgNewProvider" type="orgType" minOccurs="0"/>
```

element **orderReject/resourceReject**

diagram



type **resourceType**

properties
isRef 0
minOcc 0
maxOcc 1
content complex

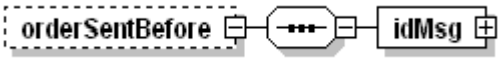
children **vehicle** **manualDescriptionResource** **idOrg** **driver** **vehiclestartLocation**

annotation
documentation
Indicates which resources that the rejection of this order are based upon.

source
```
<xs:element name="resourceReject" type="resourceType" minOccurs="0">
  <xs:annotation>
    <xs:documentation>Indicates which resources that the rejection of this order are based upon.</xs:documentation>
  </xs:annotation>
</xs:element>
```

element **orderReject/orderSentBefore**

diagram



properties

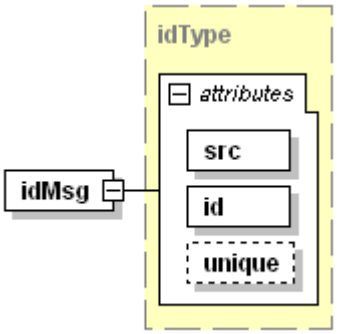
isRef	0
minOcc	0
maxOcc	1
content	complex

children **idMsg**

source
```
<xs:element name="orderSentBefore" minOccurs="0">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="idMsg" type="idType"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

element **orderReject/orderSentBefore/idMsg**

diagram



type **idType**

properties

isRef	0
content	complex

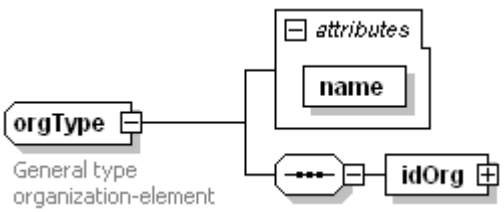
attributes

Name	Type	Use	Default	Fixed	annotation
src	**xs:string**	required			
id	**xs:string**	required			
unique	**xs:boolean**	optional	false		

source
```
<xs:element name="idMsg" type="idType"/>
```

complexType **orgType**

diagram



children **idOrg**

used by
elements **orderReject/orgNewProvider** **order/orgProvider** **SUTI/orgReceiver** **associatedReservation/orgReservation** **SUTI/orgSender**

attributes

Name	Type	Use	Default	Fixed	annotation
name	**xs:string**	required			

annotation
documentation
General type organization-element

source
```
<xs:complexType name="orgType">
 <xs:annotation>
  <xs:documentation>General type organization-element</xs:documentation>
 </xs:annotation>
 <xs:sequence>
  <xs:element name="idOrg" type="idType"/>
 </xs:sequence>
 <xs:attribute name="name" type="xs:string" use="required"/>
</xs:complexType>
```

attribute **orgType/@name**

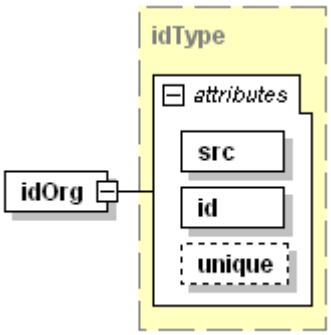
type **xs:string**

properties
isRef 0
use required

source
```
<xs:attribute name="name" type="xs:string" use="required"/>
```

element **orgType/idOrg**

diagram



type **idType**

properties
isRef 0
content complex

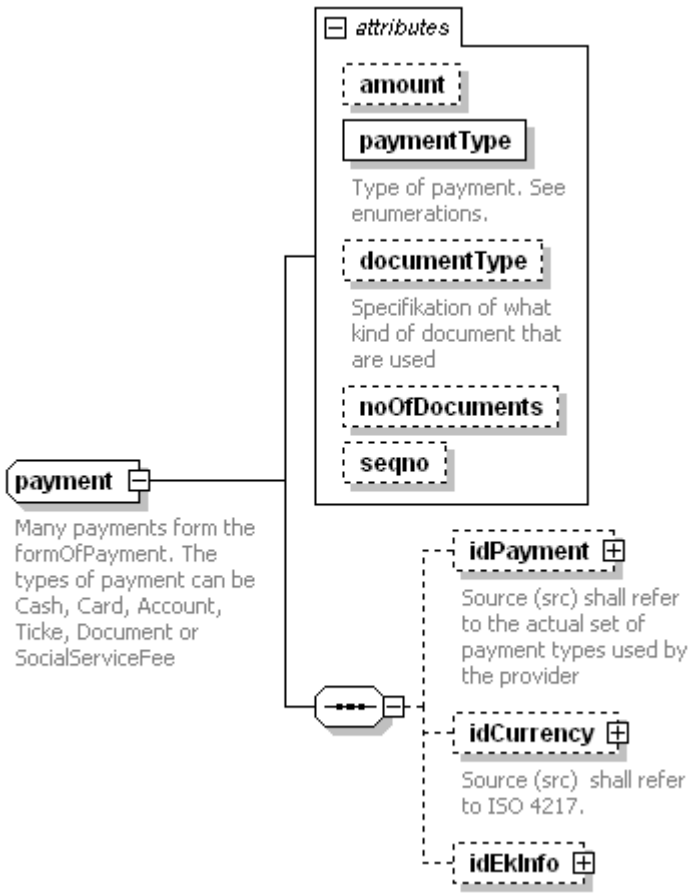
attributes

Name	Type	Use	Default	Fixed	annotation
src	**xs:string**	required			
id	**xs:string**	required			
unique	**xs:boolean**	optional	false		

source
```
<xs:element name="idOrg" type="idType"/>
```

complexType **payment**

diagram



children **idPayment** **idCurrency** **idEkInfo**

used by element **formOfPayment/payment**

attributes

Name	Type	Use	Default	Fixed	annotation
amount	**xs:float**	optional			
paymentType	**derived by: xs:string**	required			documentation Type of payment. See enumerations.
documentType	**xs:string**	optional			documentation Specifikation of what kind of document that are used
noOfDocuments	**xs:nonNegativeInteger**	optional			
seqno	**xs:positiveInteger**	optional			

annotation documentation

Many payments form the formOfPayment. The types of payment can be Cash, Card, Account, Ticke, Document or SocialServiceFee

source

```
<xs:complexType name="payment">
 <xs:annotation>
  <xs:documentation>Many payments form the formOfPayment. The types of payment can be Cash, Card, Account, Ticke, Document or SocialServiceFee</xs:documentation>
 </xs:annotation>
 <xs:sequence>
```

```
    <xs:element name="idPayment" type="idType" minOccurs="0">
      <xs:annotation>
        <xs:documentation>Source (src) shall refer to the actual set of payment types used by the
provider</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="idCurrency" type="idType" minOccurs="0">
      <xs:annotation>
        <xs:documentation>Source (src)  shall refer to ISO 4217.</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="idEkInfo" type="idEkInfo" minOccurs="0"/>
  </xs:sequence>
  <xs:attribute name="amount" type="xs:float" use="optional"/>
  <xs:attribute name="paymentType" use="required">
    <xs:annotation>
      <xs:documentation>Type of payment. See enumerations.</xs:documentation>
    </xs:annotation>
    <xs:simpleType>
      <xs:restriction base="xs:string"/>
    </xs:simpleType>
  </xs:attribute>
  <xs:attribute name="documentType" type="xs:string" use="optional">
    <xs:annotation>
      <xs:documentation>Specifikation of what kind of document that are used</xs:documentation>
    </xs:annotation>
  </xs:attribute>
  <xs:attribute name="noOfDocuments" type="xs:nonNegativeInteger" use="optional"/>
  <xs:attribute name="seqno" type="xs:positiveInteger" use="optional"/>
</xs:complexType>
```

attribute **payment/@amount**

type **xs:float**

properties
- isRef 0
- use optional

source
```
<xs:attribute name="amount" type="xs:float" use="optional"/>
```

attribute **payment/@paymentType**

type restriction of **xs:string**

properties
- isRef 0
- use required

annotation
documentation
Type of payment. See enumerations.

source
```
<xs:attribute name="paymentType" use="required">
  <xs:annotation>
    <xs:documentation>Type of payment. See enumerations.</xs:documentation>
  </xs:annotation>
  <xs:simpleType>
    <xs:restriction base="xs:string"/>
  </xs:simpleType>
</xs:attribute>
```

attribute **payment/@documentType**

type **xs:string**

properties
- isRef 0
- use optional

annotation
documentation
Specifikation of what kind of document that are used

source
```
<xs:attribute name="documentType" type="xs:string" use="optional">
  <xs:annotation>
    <xs:documentation>Specifikation of what kind of document that are used</xs:documentation>
  </xs:annotation>
</xs:attribute>
```

attribute **payment/@noOfDocuments**

type **xs:nonNegativeInteger**

properties
- isRef 0
- use optional

source
```
<xs:attribute name="noOfDocuments" type="xs:nonNegativeInteger" use="optional"/>
```

attribute **payment/@seqno**

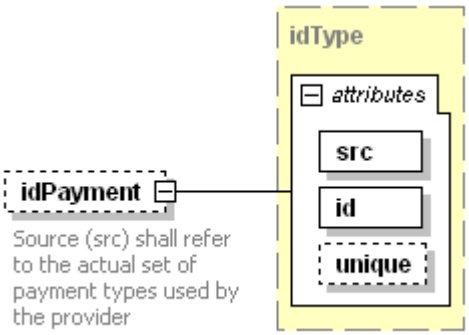
type **xs:positiveInteger**

properties
- isRef 0
- use optional

source
```
<xs:attribute name="seqno" type="xs:positiveInteger" use="optional"/>
```

element **payment/idPayment**

diagram



type **idType**

properties
- isRef 0
- minOcc 0
- maxOcc 1
- content complex

attributes

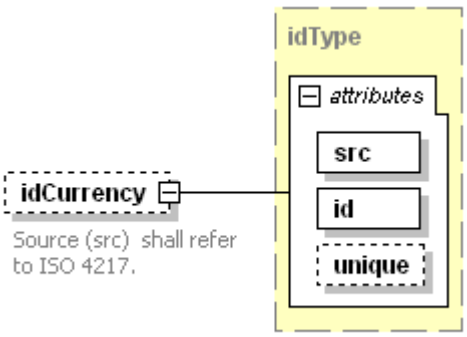
Name	Type	Use	Default	Fixed	annotation
src	**xs:string**	required			
id	**xs:string**	required			
unique	**xs:boolean**	optional	false		

annotation
documentation
Source (src) shall refer to the actual set of payment types used by the provider

source
```
<xs:element name="idPayment" type="idType" minOccurs="0">
  <xs:annotation>
    <xs:documentation>Source (src) shall refer to the actual set of payment types used by the
provider</xs:documentation>
  </xs:annotation>
</xs:element>
```

element **payment/idCurrency**

diagram



type **idType**

properties
- isRef 0
- minOcc 0
- maxOcc 1
- content complex

attributes

Name	Type	Use	Default	Fixed	annotation
src	**xs:string**	required			
id	**xs:string**	required			
unique	**xs:boolean**	optional	false		

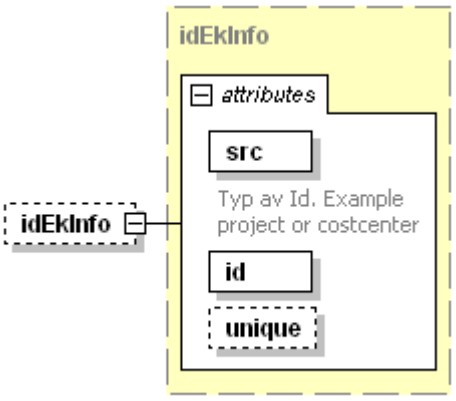
annotation
documentation
Source (src) shall refer to ISO 4217.

source

```
<xs:element name="idCurrency" type="idType" minOccurs="0">
 <xs:annotation>
  <xs:documentation>Source (src)  shall refer to ISO 4217.</xs:documentation>
 </xs:annotation>
</xs:element>
```

element **payment/idEkInfo**

diagram



type **idEkInfo**

properties
- isRef 0
- minOcc 0
- maxOcc 1
- content complex

attributes

Name	Type	Use	Default	Fixed	annotation
src	**derived by: xs:string**	required			documentation Typ av Id. Example project or

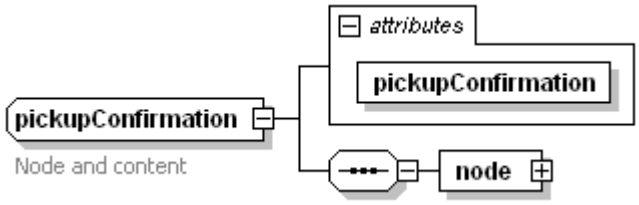
costcenter

id	**xs:string**	required		
unique	**xs:boolean**	optional	false	

source
```
<xs:element name="idEkInfo" type="idEkInfo" minOccurs="0"/>
```

complexType **pickupConfirmation**

diagram



children **node**

attributes

Name	Type	Use	Default	Fixed	annotation
pickupConfirmation	**xs:boolean**	required			

annotation
documentation
Node and content

source
```
<xs:complexType name="pickupConfirmation">
 <xs:annotation>
  <xs:documentation>Node and content</xs:documentation>
 </xs:annotation>
 <xs:sequence>
  <xs:element name="node" type="node"/>
 </xs:sequence>
 <xs:attribute name="pickupConfirmation" type="xs:boolean" use="required"/>
</xs:complexType>
```

attribute **pickupConfirmation/@pickupConfirmation**

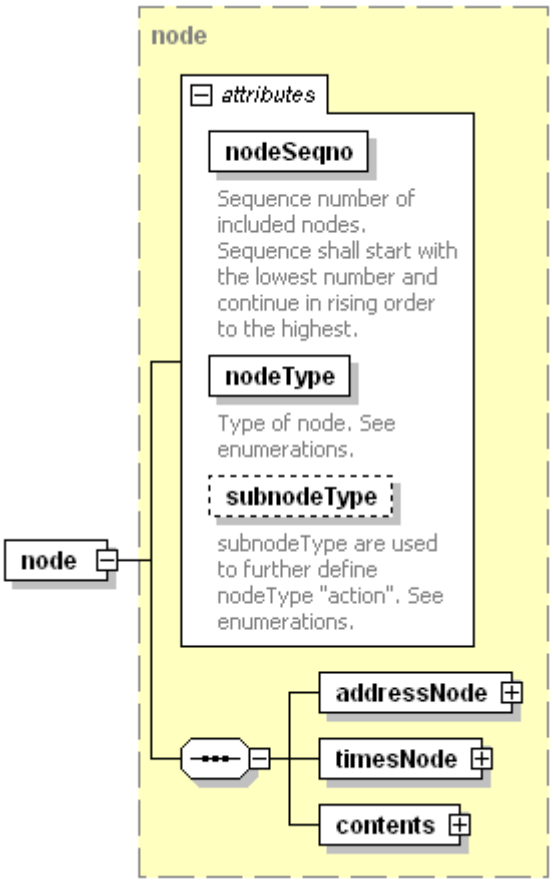
type **xs:boolean**

properties
isRef 0
use required

source
```
<xs:attribute name="pickupConfirmation" type="xs:boolean" use="required"/>
```

element **pickupConfirmation/node**

diagram



type **node**

properties
isRef 0
content complex

children **addressNode timesNode contents**

attributes

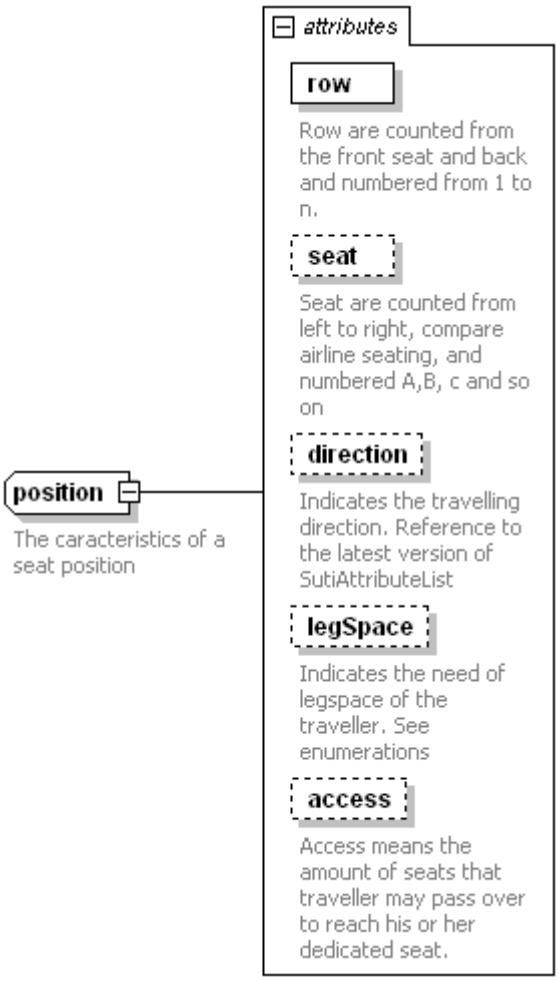
Name	Type	Use	Default	Fixed	annotation
nodeSeqno	**xs:positiveInteger**	required			documentation Sequence number of included nodes. Sequence shall start with the lowest number and continue in rising order to the highest.
nodeType	**derived by: xs:string**	required			documentation Type of node. See enumerations.
subnodeType	**derived by: xs:string**	optional			documentation subnodeType are used to further define nodeType

"action". See enumerations.

source <xs:element name="node" type="node"/>

complexType **position**

diagram



used by element **seats/position**

attributes

Name	Type	Use	Default	Fixed	annotation
row	**xs:positiveInteger**	required			documentation Row are counted from the front seat and back and numbered from 1 to n.
seat	**xs:string**	optional			documentation Seat are counted from left to right, compare airline seating, and numbered A,B, c and so on
direction	**derived by: xs:string**	optional			documentation

			Indicates the travelling direction. Reference to the latest version of SutiAttributeList
legSpace	**derived by: xs:string**	optional	documentation Indicates the need of legspace of the traveller. See enumerations
access	**xs:nonNegativeInteger**	optional	documentation Access means the amount of seats that traveller may pass over to reach his or her dedicated seat.

annotation
documentation
The caracteristics of a seat position

source

```
<xs:complexType name="position">
 <xs:annotation>
  <xs:documentation>The caracteristics of a seat position</xs:documentation>
 </xs:annotation>
 <xs:attribute name="row" type="xs:positiveInteger" use="required">
  <xs:annotation>
   <xs:documentation>Row are counted from the front seat and back and numbered from 1 to
n.</xs:documentation>
  </xs:annotation>
 </xs:attribute>
 <xs:attribute name="seat" type="xs:string" use="optional">
  <xs:annotation>
   <xs:documentation>Seat are counted from left to right, compare airline seating, and numbered A,B, c
and so on</xs:documentation>
  </xs:annotation>
 </xs:attribute>
 <xs:attribute name="direction" use="optional">
  <xs:annotation>
   <xs:documentation>Indicates the travelling direction. Reference to the latest version of
SutiAttributeList</xs:documentation>
  </xs:annotation>
  <xs:simpleType>
   <xs:restriction base="xs:string"/>
  </xs:simpleType>
 </xs:attribute>
 <xs:attribute name="legSpace" use="optional">
  <xs:annotation>
   <xs:documentation>Indicates the need of legspace of the traveller. See
enumerations</xs:documentation>
  </xs:annotation>
  <xs:simpleType>
   <xs:restriction base="xs:string"/>
  </xs:simpleType>
 </xs:attribute>
 <xs:attribute name="access" type="xs:nonNegativeInteger" use="optional">
  <xs:annotation>
   <xs:documentation>Access means the amount of seats that traveller may pass over to reach his or her
dedicated seat. </xs:documentation>
```

```
    </xs:annotation>
  </xs:attribute>
</xs:complexType>
```

attribute **position/@row**

type **xs:positiveInteger**

properties
- isRef 0
- use required

annotation
documentation
Row are counted from the front seat and back and numbered from 1 to n.

source

```
<xs:attribute name="row" type="xs:positiveInteger" use="required">
  <xs:annotation>
    <xs:documentation>Row are counted from the front seat and back and numbered from 1 to
n.</xs:documentation>
  </xs:annotation>
</xs:attribute>
```

attribute **position/@seat**

type **xs:string**

properties
- isRef 0
- use optional

annotation
documentation
Seat are counted from left to right, compare airline seating, and numbered A,B, c and so on

source

```
<xs:attribute name="seat" type="xs:string" use="optional">
  <xs:annotation>
    <xs:documentation>Seat are counted from left to right, compare airline seating, and numbered A,B, c
and so on</xs:documentation>
  </xs:annotation>
</xs:attribute>
```

attribute **position/@direction**

type restriction of **xs:string**

properties
- isRef 0
- use optional

annotation
documentation
Indicates the travelling direction. Reference to the latest version of SutiAttributeList

source

```
<xs:attribute name="direction" use="optional">
  <xs:annotation>
    <xs:documentation>Indicates the travelling direction. Reference to the latest version of
SutiAttributeList</xs:documentation>
  </xs:annotation>
  <xs:simpleType>
    <xs:restriction base="xs:string"/>
  </xs:simpleType>
</xs:attribute>
```

attribute **position/@legSpace**

type restriction of **xs:string**

properties
- isRef 0
- use optional

annotation
documentation
Indicates the need of legspace of the traveller. See enumerations

source

```
<xs:attribute name="legSpace" use="optional">
  <xs:annotation>
    <xs:documentation>Indicates the need of legspace of the traveller. See
```

```
enumerations</xs:documentation>
  </xs:annotation>
  <xs:simpleType>
    <xs:restriction base="xs:string"/>
  </xs:simpleType>
</xs:attribute>
```

attribute **position/@access**

type **xs:nonNegativeInteger**

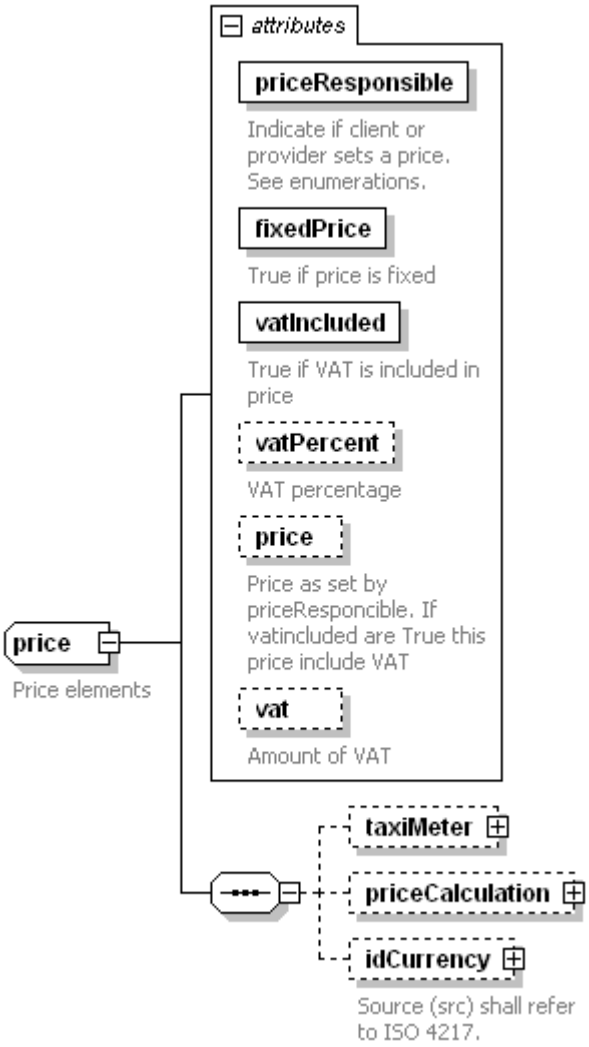
properties
isRef 0
use optional

annotation
documentation
Access means the amount of seats that traveller may pass over to reach his or her dedicated seat.

source
```
<xs:attribute name="access" type="xs:nonNegativeInteger" use="optional">
  <xs:annotation>
    <xs:documentation>Access means the amount of seats that traveller may pass over to reach his or her dedicated seat. </xs:documentation>
  </xs:annotation>
</xs:attribute>
```

complexType **price**

diagram



children **taxiMeter** **priceCalculation** **idCurrency**

used by element **economyType/price**

attributes

Name	Type	Use	Default	Fixed	annotation
priceResponsible	**derived by: xs:string**	required			documentation Indicate if client or provider sets a price. See enumerations.
fixedPrice	**xs:boolean**	required			documentation True if price is fixed
vatIncluded	**xs:boolean**	required			documentation True if VAT is included in price
vatPercent	**xs:float**	optional			documentation VAT percentage
price	**xs:float**	optional			documentation Price as set by priceResponcible. If vatincluded are True this price include VAT
vat	**xs:float**	optional			documentation Amount of VAT

annotation documentation
Price elements

source

```
<xs:complexType name="price">
 <xs:annotation>
  <xs:documentation>Price elements</xs:documentation>
 </xs:annotation>
 <xs:sequence>
  <xs:element name="taxiMeter" type="taxiMeter" minOccurs="0"/>
  <xs:element name="priceCalculation" type="priceCalculation" minOccurs="0"/>
  <xs:element name="idCurrency" type="idType" minOccurs="0">
   <xs:annotation>
    <xs:documentation>Source (src) shall refer to ISO 4217.</xs:documentation>
   </xs:annotation>
  </xs:element>
 </xs:sequence>
 <xs:attribute name="priceResponsible" use="required">
  <xs:annotation>
   <xs:documentation>Indicate if client or provider sets a price. See enumerations.</xs:documentation>
  </xs:annotation>
  <xs:simpleType>
   <xs:restriction base="xs:string"/>
  </xs:simpleType>
 </xs:attribute>
 <xs:attribute name="fixedPrice" type="xs:boolean" use="required">
  <xs:annotation>
   <xs:documentation>True if price is fixed</xs:documentation>
  </xs:annotation>
 </xs:attribute>
 <xs:attribute name="vatIncluded" type="xs:boolean" use="required">
  <xs:annotation>
```

```
      <xs:documentation>True if VAT is included in price</xs:documentation>
    </xs:annotation>
  </xs:attribute>
  <xs:attribute name="vatPercent" type="xs:float" use="optional">
    <xs:annotation>
      <xs:documentation>VAT percentage</xs:documentation>
    </xs:annotation>
  </xs:attribute>
  <xs:attribute name="price" type="xs:float" use="optional">
    <xs:annotation>
      <xs:documentation>Price as set by priceResponcible. If vatincluded are True this price include
VAT</xs:documentation>
    </xs:annotation>
  </xs:attribute>
  <xs:attribute name="vat" type="xs:float" use="optional">
    <xs:annotation>
      <xs:documentation>Amount of VAT</xs:documentation>
    </xs:annotation>
  </xs:attribute>
</xs:complexType>
```

attribute **price/@priceResponsible**

type restriction of **xs:string**

properties
- isRef 0
- use required

annotation
documentation
Indicate if client or provider sets a price. See enumerations.

source

```
<xs:attribute name="priceResponsible" use="required">
  <xs:annotation>
    <xs:documentation>Indicate if client or provider sets a price. See enumerations.</xs:documentation>
  </xs:annotation>
  <xs:simpleType>
    <xs:restriction base="xs:string"/>
  </xs:simpleType>
</xs:attribute>
```

attribute **price/@fixedPrice**

type **xs:boolean**

properties
- isRef 0
- use required

annotation
documentation
True if price is fixed

source

```
<xs:attribute name="fixedPrice" type="xs:boolean" use="required">
  <xs:annotation>
    <xs:documentation>True if price is fixed</xs:documentation>
  </xs:annotation>
</xs:attribute>
```

attribute **price/@vatIncluded**

type **xs:boolean**

properties
- isRef 0
- use required

annotation
documentation
True if VAT is included in price

source

```
<xs:attribute name="vatIncluded" type="xs:boolean" use="required">
  <xs:annotation>
    <xs:documentation>True if VAT is included in price</xs:documentation>
  </xs:annotation>
```

```
</xs:attribute>
```

attribute **price/@vatPercent**

type **xs:float**

properties
isRef 0
use optional

annotation
documentation
VAT percentage

source

```
<xs:attribute name="vatPercent" type="xs:float" use="optional">
 <xs:annotation>
  <xs:documentation>VAT percentage</xs:documentation>
 </xs:annotation>
</xs:attribute>
```

attribute **price/@price**

type **xs:float**

properties
isRef 0
use optional

annotation
documentation
Price as set by priceResponcible. If vatincluded are True this price include VAT

source

```
<xs:attribute name="price" type="xs:float" use="optional">
 <xs:annotation>
  <xs:documentation>Price as set by priceResponcible. If vatincluded are True this price include
VAT</xs:documentation>
 </xs:annotation>
</xs:attribute>
```

attribute **price/@vat**

type **xs:float**

properties
isRef 0
use optional

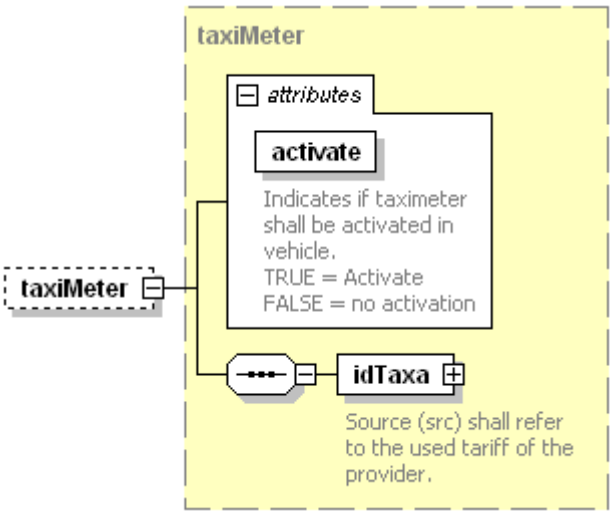
annotation
documentation
Amount of VAT

source

```
<xs:attribute name="vat" type="xs:float" use="optional">
 <xs:annotation>
  <xs:documentation>Amount of VAT</xs:documentation>
 </xs:annotation>
</xs:attribute>
```

element **price/taxiMeter**

diagram



type **taxiMeter**

properties
- isRef 0
- minOcc 0
- maxOcc 1
- content complex

children **idTaxa**

attributes

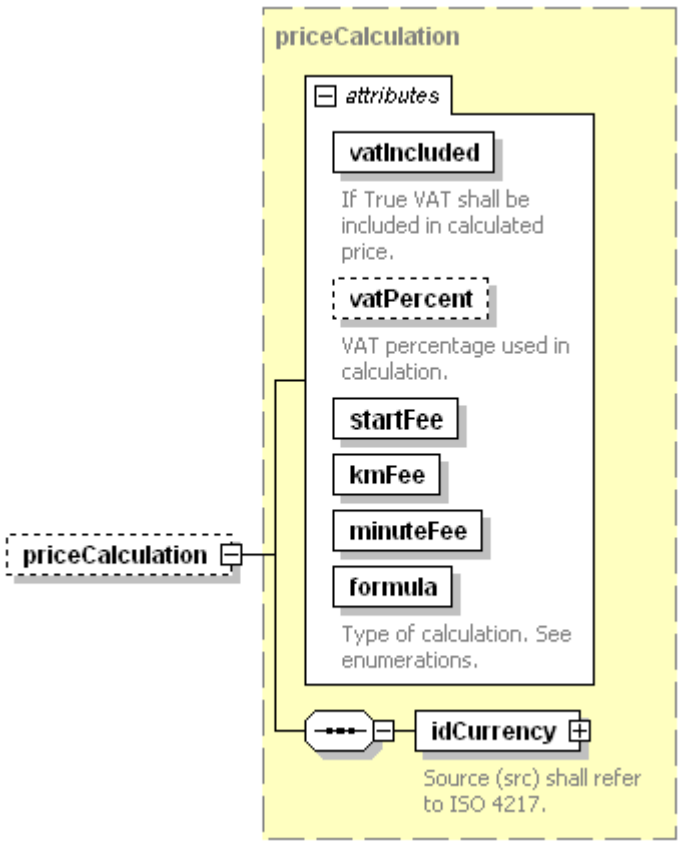
Name	Type	Use	Default	Fixed	annotation
activate	**xs:boolean**	required			documentation: Indicates if taximeter shall be activated in vehicle. TRUE = Activate FALSE = no activation

source

```
<xs:element name="taxiMeter" type="taxiMeter" minOccurs="0"/>
```

element **price/priceCalculation**

diagram



type **priceCalculation**

properties

isRef	0
minOcc	0
maxOcc	1
content	complex

children **idCurrency**

attributes

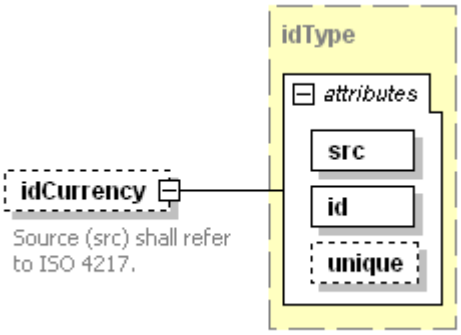
Name	Type	Use	Default	Fixed	annotation
vatIncluded	**xs:boolean**	required			documentation: If True VAT shall be included in calculated price.
vatPercent	**xs:float**	optional			documentation: VAT percentage used in calculation.
startFee	**xs:float**	required			
kmFee	**xs:float**	required			
minuteFee	**xs:float**	required			
formula	**derived by: xs:string**	required			documentation: Type of calculation. See enumerations.

source

```
<xs:element name="priceCalculation" type="priceCalculation" minOccurs="0"/>
```

element **price/idCurrency**

diagram



type **idType**

properties

isRef	0
minOcc	0
maxOcc	1
content	complex

attributes

Name	Type	Use	Default	Fixed	annotation
src	**xs:string**	required			
id	**xs:string**	required			
unique	**xs:boolean**	optional	false		

annotation

documentation

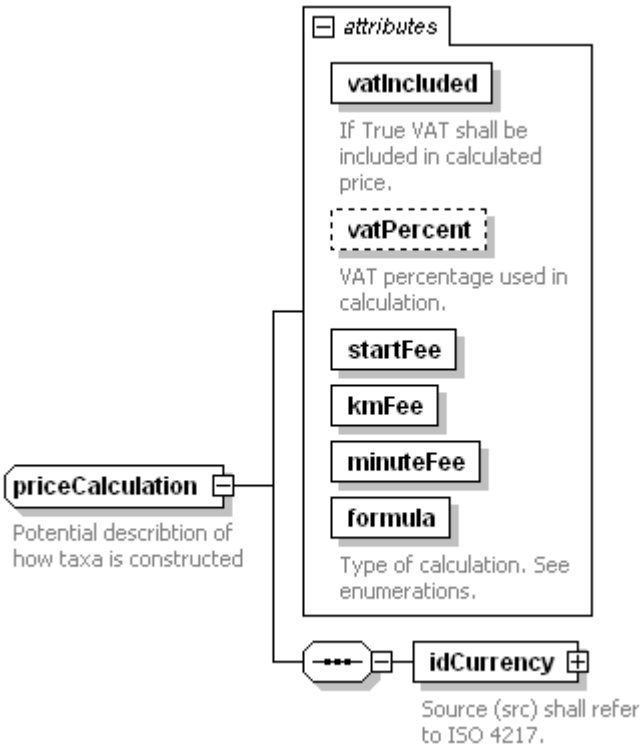
Source (src) shall refer to ISO 4217.

source

```
<xs:element name="idCurrency" type="idType" minOccurs="0">
 <xs:annotation>
  <xs:documentation>Source (src) shall refer to ISO 4217.</xs:documentation>
 </xs:annotation>
</xs:element>
```

complexType **priceCalculation**

diagram



children **idCurrency**

used by element **price/priceCalculation**

attributes

Name	Type	Use	Default	Fixed	annotation
vatIncluded	**xs:boolean**	required			documentation If True VAT shall be included in calculated price.
vatPercent	**xs:float**	optional			documentation VAT percentage used in calculation.
startFee	**xs:float**	required			
kmFee	**xs:float**	required			
minuteFee	**xs:float**	required			
formula	**derived by: xs:string**	required			documentation Type of calculation. See enumerations.

annotation documentation
Potential describtion of how taxa is constructed

source

```
<xs:complexType name="priceCalculation">
 <xs:annotation>
  <xs:documentation>Potential describtion of how taxa is constructed</xs:documentation>
 </xs:annotation>
 <xs:sequence>
  <xs:element name="idCurrency" type="idType">
   <xs:annotation>
    <xs:documentation>Source (src) shall refer to ISO 4217.</xs:documentation>
   </xs:annotation>
  </xs:element>
 </xs:sequence>
 <xs:attribute name="vatIncluded" type="xs:boolean" use="required">
  <xs:annotation>
   <xs:documentation>If True VAT shall be included in calculated price.</xs:documentation>
  </xs:annotation>
 </xs:attribute>
 <xs:attribute name="vatPercent" type="xs:float" use="optional">
  <xs:annotation>
   <xs:documentation>VAT percentage used in calculation.</xs:documentation>
  </xs:annotation>
 </xs:attribute>
 <xs:attribute name="startFee" type="xs:float" use="required"/>
 <xs:attribute name="kmFee" type="xs:float" use="required"/>
 <xs:attribute name="minuteFee" type="xs:float" use="required"/>
 <xs:attribute name="formula" use="required">
  <xs:annotation>
   <xs:documentation>Type of calculation. See enumerations.</xs:documentation>
  </xs:annotation>
  <xs:simpleType>
   <xs:restriction base="xs:string"/>
  </xs:simpleType>
 </xs:attribute>
</xs:complexType>
```

attribute **priceCalculation/@vatIncluded**

type **xs:boolean**

properties
  isRef 0
  use required

annotation
  documentation
  If True VAT shall be included in calculated price.

source
```
<xs:attribute name="vatIncluded" type="xs:boolean" use="required">
 <xs:annotation>
  <xs:documentation>If True VAT shall be included in calculated price.</xs:documentation>
 </xs:annotation>
</xs:attribute>
```

attribute **priceCalculation/@vatPercent**

type **xs:float**

properties
  isRef 0
  use optional

annotation
  documentation
  VAT percentage used in calculation.

source
```
<xs:attribute name="vatPercent" type="xs:float" use="optional">
 <xs:annotation>
  <xs:documentation>VAT percentage used in calculation.</xs:documentation>
 </xs:annotation>
</xs:attribute>
```

attribute **priceCalculation/@startFee**

type **xs:float**

properties
  isRef 0
  use required

source
```
<xs:attribute name="startFee" type="xs:float" use="required"/>
```

attribute **priceCalculation/@kmFee**

type **xs:float**

properties
  isRef 0
  use required

source
```
<xs:attribute name="kmFee" type="xs:float" use="required"/>
```

attribute **priceCalculation/@minuteFee**

type **xs:float**

properties
  isRef 0
  use required

source
```
<xs:attribute name="minuteFee" type="xs:float" use="required"/>
```

attribute **priceCalculation/@formula**

type restriction of **xs:string**

properties
  isRef 0
  use required

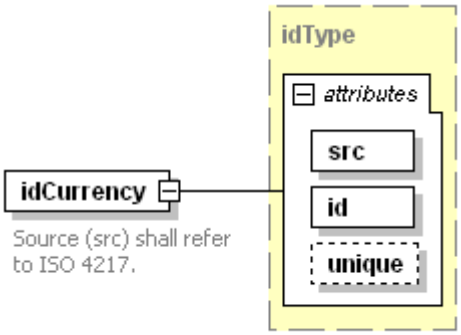
annotation
  documentation
  Type of calculation. See enumerations.

source
```
<xs:attribute name="formula" use="required">
 <xs:annotation>
  <xs:documentation>Type of calculation. See enumerations.</xs:documentation>
 </xs:annotation>
 <xs:simpleType>
```

```
    <xs:restriction base="xs:string"/>
  </xs:simpleType>
</xs:attribute>
```

element **priceCalculation/idCurrency**

diagram



type **idType**

properties
- isRef 0
- content complex

attributes

Name	Type	Use	Default	Fixed	annotation
src	**xs:string**	required			
id	**xs:string**	required			
unique	**xs:boolean**	optional	false		

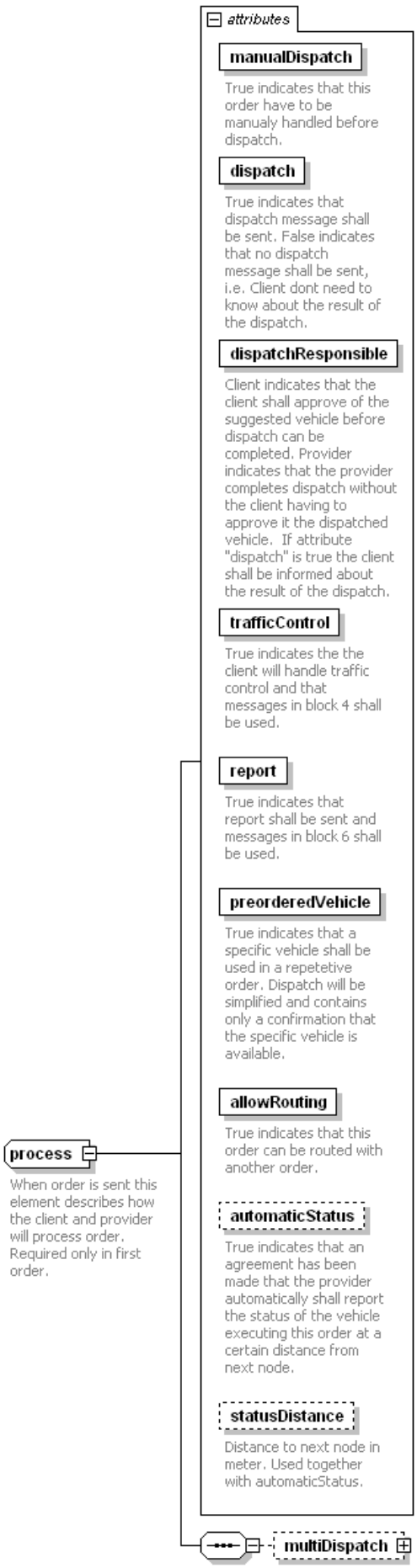
annotation
documentation
Source (src) shall refer to ISO 4217.

source

```
<xs:element name="idCurrency" type="idType">
  <xs:annotation>
    <xs:documentation>Source (src) shall refer to ISO 4217.</xs:documentation>
  </xs:annotation>
</xs:element>
```

complexType **process**

diagram


process
When order is sent this element describes how the client and provider will process order. Required only in first order.
attributes
manualDispatch
True indicates that this order have to be manualy handled before dispatch.
dispatch
True indicates that dispatch message shall be sent. False indicates that no dispatch message shall be sent, i.e. Client dont need to know about the result of the dispatch.
dispatchResponsible
Client indicates that the client shall approve of the suggested vehicle before dispatch can be completed. Provider indicates that the provider completes dispatch without the client having to approve it the dispatched vehicle. If attribute "dispatch" is true the client shall be informed about the result of the dispatch.
trafficControl
True indicates the the client will handle traffic control and that messages in block 4 shall be used.
report
True indicates that report shall be sent and messages in block 6 shall be used.
preorderedVehicle
True indicates that a specific vehicle shall be used in a repetetive order. Dispatch will be simplified and contains only a confirmation that the specific vehicle is available.
allowRouting
True indicates that this order can be routed with another order.
automaticStatus
True indicates that an agreement has been made that the provider automatically shall report the status of the vehicle executing this order at a certain distance from next node.
statusDistance
Distance to next node in meter. Used together with automaticStatus.
multiDispatch

children **multiDispatch**

used by element **order/process**

attributes

Name	Type	Use	Default	Fixed	annotation
manualDispatch	**xs:boolean**	required			documentation True indicates that this order have to be manualy handled before dispatch.
dispatch	**xs:boolean**	required			documentation True indicates that dispatch message shall be sent. False indicates that no dispatch message shall be sent, i.e. Client dont need to know about the result of the dispatch.
dispatchResponsible	**derived by: xs:string**	required			documentation Client indicates that the client shall approve of the suggested vehicle before dispatch can be completed. Provider indicates that the provider completes dispatch without the client having to approve it the dispatched vehicle. If attribute "dispatch" is true the client shall be informed about the result of the dispatch.
trafficControl	**xs:boolean**	required			documentation True indicates the the client will handle traffic control and that messages in block 4 shall be used.

	report	**xs:boolean**	required	documentation True indicates that report shall be sent and messages in block 6 shall be used.
	preorderedVehicle	**xs:boolean**	required	documentation True indicates that a specific vehicle shall be used in a repetetive order. Dispatch will be simplified and contains only a confirmation that the specific vehicle is available.
	allowRouting	**xs:boolean**	required	documentation True indicates that this order can be routed with another order.
	automaticStatus	**xs:boolean**	optional	documentation True indicates that an agreement has been made that the provider automatically shall report the status of the vehicle executing this order at a certain distance from next node.
	statusDistance	**xs:nonNegativeInteger**	optional	documentation Distance to next node in meter. Used together with automaticStatus.
annotation	documentation When order is sent this element describes how the client and provider will process order. Required only in			

first order.

source
```
<xs:complexType name="process">
 <xs:annotation>
  <xs:documentation>When order is sent this element describes how the client and provider will process order. Required only in first order.</xs:documentation>
 </xs:annotation>
 <xs:sequence>
  <xs:element name="multiDispatch" type="multiDispatch" minOccurs="0"/>
 </xs:sequence>
 <xs:attribute name="manualDispatch" type="xs:boolean" use="required">
  <xs:annotation>
   <xs:documentation>True indicates that this order have to be manualy handled before dispatch.</xs:documentation>
  </xs:annotation>
 </xs:attribute>
 <xs:attribute name="dispatch" type="xs:boolean" use="required">
  <xs:annotation>
   <xs:documentation>True indicates that dispatch message shall be sent. False indicates that no dispatch message shall be sent, i.e. Client dont need to know about the result of the dispatch.</xs:documentation>
  </xs:annotation>
 </xs:attribute>
 <xs:attribute name="dispatchResponsible" use="required">
  <xs:annotation>
   <xs:documentation>Client indicates that the client shall approve of the suggested vehicle before dispatch can be completed. Provider indicates that the provider completes dispatch without the client having to approve it the dispatched vehicle.  If attribute "dispatch" is true the client shall be informed about the result of the dispatch.</xs:documentation>
  </xs:annotation>
  <xs:simpleType>
   <xs:restriction base="xs:string"/>
  </xs:simpleType>
 </xs:attribute>
 <xs:attribute name="trafficControl" type="xs:boolean" use="required">
  <xs:annotation>
   <xs:documentation>True indicates the the client will handle traffic control and that messages in block 4 shall be used.
</xs:documentation>
  </xs:annotation>
 </xs:attribute>
 <xs:attribute name="report" type="xs:boolean" use="required">
  <xs:annotation>
   <xs:documentation>True indicates that report shall be sent and messages in block 6 shall be used.
</xs:documentation>
  </xs:annotation>
 </xs:attribute>
 <xs:attribute name="preorderedVehicle" type="xs:boolean" use="required">
  <xs:annotation>
   <xs:documentation>True indicates that a specific vehicle shall be used in a repetetive order. Dispatch will be simplified and contains only a confirmation that the specific vehicle is available.
</xs:documentation>
  </xs:annotation>
 </xs:attribute>
 <xs:attribute name="allowRouting" type="xs:boolean" use="required">
  <xs:annotation>
   <xs:documentation>True indicates that this order can be routed with another order.
</xs:documentation>
  </xs:annotation>
 </xs:attribute>
 <xs:attribute name="automaticStatus" type="xs:boolean" use="optional">
  <xs:annotation>
   <xs:documentation>True indicates that an agreement has been made that the provider automatically shall report the status of the vehicle executing this order at a certain distance from next node.
</xs:documentation>
  </xs:annotation>
 </xs:attribute>
 <xs:attribute name="statusDistance" type="xs:nonNegativeInteger" use="optional">
```

```
    <xs:annotation>
      <xs:documentation>Distance to next node in meter. Used together with automaticStatus.
</xs:documentation>
    </xs:annotation>
  </xs:attribute>
</xs:complexType>
```

attribute **process/@manualDispatch**

type **xs:boolean**

properties
- isRef 0
- use required

annotation
documentation
True indicates that this order have to be manualy handled before dispatch.

source

```
<xs:attribute name="manualDispatch" type="xs:boolean" use="required">
  <xs:annotation>
    <xs:documentation>True indicates that this order have to be manualy handled before
dispatch.</xs:documentation>
  </xs:annotation>
</xs:attribute>
```

attribute **process/@dispatch**

type **xs:boolean**

properties
- isRef 0
- use required

annotation
documentation
True indicates that dispatch message shall be sent. False indicates that no dispatch message shall be sent, i.e. Client dont need to know about the result of the dispatch.

source

```
<xs:attribute name="dispatch" type="xs:boolean" use="required">
  <xs:annotation>
    <xs:documentation>True indicates that dispatch message shall be sent. False indicates that no dispatch
message shall be sent, i.e. Client dont need to know about the result of the dispatch.</xs:documentation>
  </xs:annotation>
</xs:attribute>
```

attribute **process/@dispatchResponsible**

type restriction of **xs:string**

properties
- isRef 0
- use required

annotation
documentation
Client indicates that the client shall approve of the suggested vehicle before dispatch can be completed. Provider indicates that the provider completes dispatch without the client having to approve it the dispatched vehicle. If attribute "dispatch" is true the client shall be informed about the result of the dispatch.

source

```
<xs:attribute name="dispatchResponsible" use="required">
  <xs:annotation>
    <xs:documentation>Client indicates that the client shall approve of the suggested vehicle before
dispatch can be completed. Provider indicates that the provider completes dispatch without the client
having to approve it the dispatched vehicle.  If attribute "dispatch" is true the client shall be informed about
the result of the dispatch.</xs:documentation>
  </xs:annotation>
  <xs:simpleType>
    <xs:restriction base="xs:string"/>
  </xs:simpleType>
</xs:attribute>
```

attribute **process/@trafficControl**

type **xs:boolean**

properties
isRef 0
use required

annotation
documentation
True indicates the the client will handle traffic control and that messages in block 4 shall be used.

source
```
<xs:attribute name="trafficControl" type="xs:boolean" use="required">
 <xs:annotation>
  <xs:documentation>True indicates the the client will handle traffic control and that messages in block 4 shall be used.
</xs:documentation>
 </xs:annotation>
</xs:attribute>
```

attribute **process/@report**

type **xs:boolean**

properties
isRef 0
use required

annotation
documentation
True indicates that report shall be sent and messages in block 6 shall be used.

source
```
<xs:attribute name="report" type="xs:boolean" use="required">
 <xs:annotation>
  <xs:documentation>True indicates that report shall be sent and messages in block 6 shall be used.
</xs:documentation>
 </xs:annotation>
</xs:attribute>
```

attribute **process/@preorderedVehicle**

type **xs:boolean**

properties
isRef 0
use required

annotation
documentation
True indicates that a specific vehicle shall be used in a repetetive order. Dispatch will be simplified and contains only a confirmation that the specific vehicle is available.

source
```
<xs:attribute name="preorderedVehicle" type="xs:boolean" use="required">
 <xs:annotation>
  <xs:documentation>True indicates that a specific vehicle shall be used in a repetetive order. Dispatch will be simplified and contains only a confirmation that the specific vehicle is available.
</xs:documentation>
 </xs:annotation>
</xs:attribute>
```

attribute **process/@allowRouting**

type **xs:boolean**

properties
isRef 0
use required

annotation
documentation
True indicates that this order can be routed with another order.

source
```
<xs:attribute name="allowRouting" type="xs:boolean" use="required">
```

```
  <xs:annotation>
    <xs:documentation>True indicates that this order can be routed with another order.
</xs:documentation>
  </xs:annotation>
</xs:attribute>
```

attribute **process/@automaticStatus**

type **xs:boolean**

properties
isRef 0
use optional

annotation
documentation
True indicates that an agreement has been made that the provider automatically shall report the status of the vehicle executing this order at a certain distance from next node.

source

```
<xs:attribute name="automaticStatus" type="xs:boolean" use="optional">
  <xs:annotation>
    <xs:documentation>True indicates that an agreement has been made that the provider automatically
shall report the status of the vehicle executing this order at a certain distance from next node.
</xs:documentation>
  </xs:annotation>
</xs:attribute>
```

attribute **process/@statusDistance**

type **xs:nonNegativeInteger**

properties
isRef 0
use optional

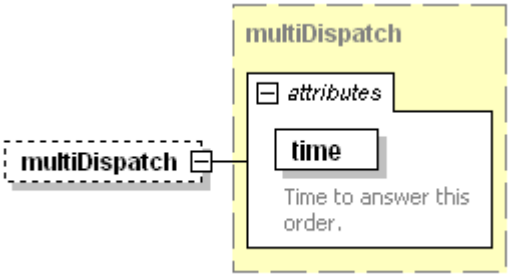
annotation
documentation
Distance to next node in meter. Used together with automaticStatus.

source

```
<xs:attribute name="statusDistance" type="xs:nonNegativeInteger" use="optional">
  <xs:annotation>
    <xs:documentation>Distance to next node in meter. Used together with automaticStatus.
</xs:documentation>
  </xs:annotation>
</xs:attribute>
```

element **process/multiDispatch**

diagram



type **multiDispatch**

properties
isRef 0
minOcc 0
maxOcc 1
content complex

attributes

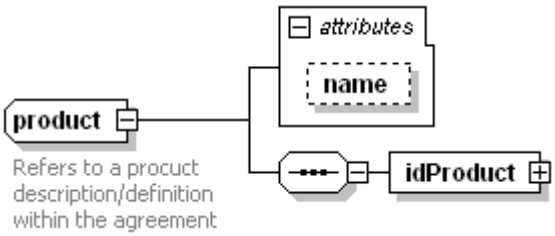
Name	Type	Use	Default	Fixed	annotation
time	**xs:dateTime**	required			documentation Time to

answer this order.

source `<xs:element name="multiDispatch" type="multiDispatch" minOccurs="0"/>`

complexType **product**

diagram



children **idProduct**

used by element **agreement/product**

attributes

Name	Type	Use	Default	Fixed	annotation
name	**xs:string**	optional			

annotation documentation
Refers to a procuct description/definition within the agreement

source
```
<xs:complexType name="product">
 <xs:annotation>
  <xs:documentation>Refers to a procuct description/definition within the agreement</xs:documentation>
 </xs:annotation>
 <xs:sequence>
  <xs:element name="idProduct" type="idType"/>
 </xs:sequence>
 <xs:attribute name="name" type="xs:string" use="optional"/>
</xs:complexType>
```

attribute **product/@name**

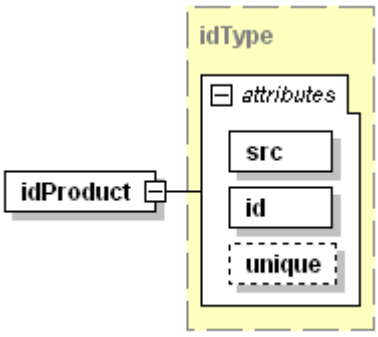
type **xs:string**

properties
isRef 0
use optional

source `<xs:attribute name="name" type="xs:string" use="optional"/>`

element **product/idProduct**

diagram



type **idType**

properties
isRef 0
content complex

attributes

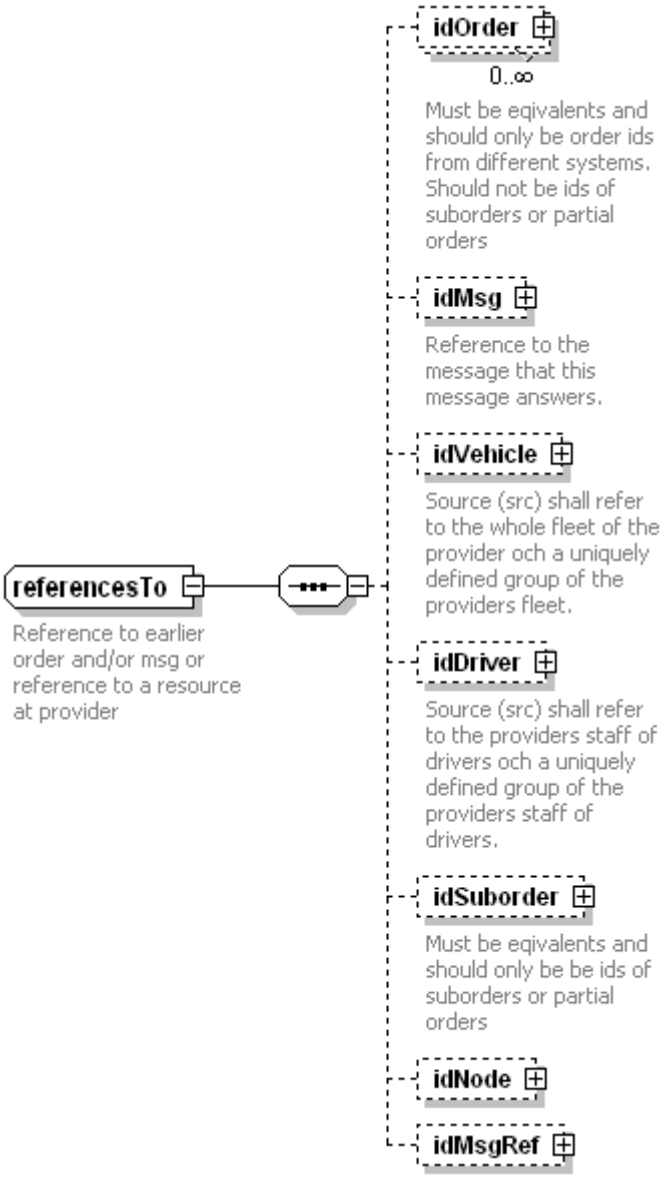
Name	Type	Use	Default	Fixed	annotation
src	**xs:string**	required			

	id	**xs:string**	required	
	unique	**xs:boolean**	optional	false

source
```
<xs:element name="idProduct" type="idType"/>
```

complexType **referencesTo**

diagram



children **idOrder** **idMsg** **idVehicle** **idDriver** **idSuborder** **idNode** **idMsgRef**

used by element **msg/referencesTo**

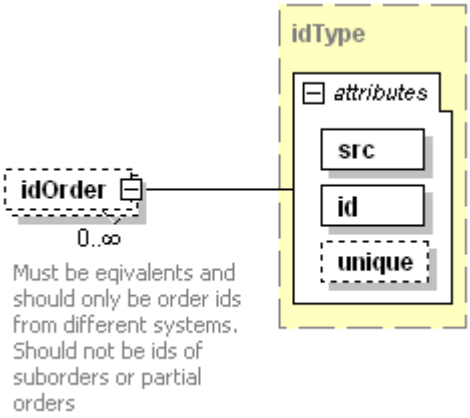
annotation documentation
Reference to earlier order and/or msg or reference to a resource at provider

source
```
<xs:complexType name="referencesTo">
 <xs:annotation>
  <xs:documentation>Reference to earlier order and/or msg or reference to a resource at provider</xs:documentation>
 </xs:annotation>
 <xs:sequence>
  <xs:element name="idOrder" type="idType" minOccurs="0" maxOccurs="unbounded">
   <xs:annotation>
    <xs:documentation>Must be eqivalents and should only be order ids from different systems. Should not be ids of suborders or partial orders</xs:documentation>
```

```
    </xs:annotation>
   </xs:element>
   <xs:element name="idMsg" type="idType" minOccurs="0">
    <xs:annotation>
     <xs:documentation>Reference to the message that this message answers.</xs:documentation>
    </xs:annotation>
   </xs:element>
   <xs:element name="idVehicle" type="idType" minOccurs="0">
    <xs:annotation>
     <xs:documentation>Source (src) shall refer to the whole fleet of the provider och a uniquely defined
group of the providers fleet.</xs:documentation>
    </xs:annotation>
   </xs:element>
   <xs:element name="idDriver" type="idType" minOccurs="0">
    <xs:annotation>
     <xs:documentation>Source (src) shall refer to the providers staff of drivers och a uniquely defined
group of the providers staff of drivers.</xs:documentation>
    </xs:annotation>
   </xs:element>
   <xs:element name="idSuborder" type="idType" minOccurs="0">
    <xs:annotation>
     <xs:documentation>Must be eqivalents and should only be be ids of suborders or partial
orders</xs:documentation>
    </xs:annotation>
   </xs:element>
   <xs:element name="idNode" type="idType" minOccurs="0"/>
   <xs:element name="idMsgRef" type="idMsgRef " minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
```

element **referencesTo/idOrder**

diagram



type **idType**

properties

isRef	0
minOcc	0
maxOcc	unbounded
content	complex

attributes

Name	Type	Use	Default	Fixed	annotation
src	**xs:string**	required			
id	**xs:string**	required			
unique	**xs:boolean**	optional	false		

annotation

documentation

Must be eqivalents and should only be order ids from different systems. Should not be ids of suborders or partial orders

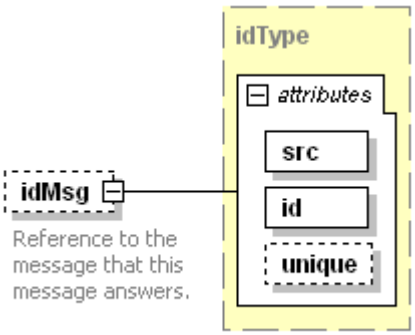
source

```
<xs:element name="idOrder" type="idType" minOccurs="0" maxOccurs="unbounded">
 <xs:annotation>
  <xs:documentation>Must be eqivalents and should only be order ids from different systems. Should not
be ids of suborders or partial orders</xs:documentation>
 </xs:annotation>
```

```
</xs:element>
```

element **referencesTo/idMsg**

diagram



type **idType**

properties

isRef	0
minOcc	0
maxOcc	1
content	complex

attributes

Name	Type	Use	Default	Fixed	annotation
src	**xs:string**	required			
id	**xs:string**	required			
unique	**xs:boolean**	optional	false		

annotation

documentation

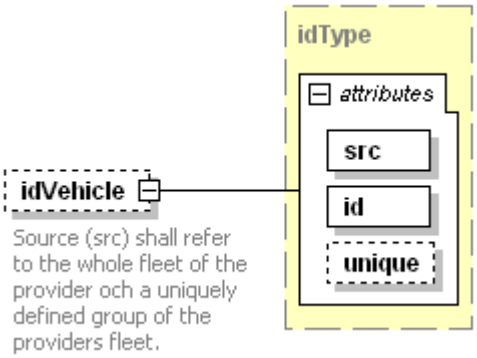
Reference to the message that this message answers.

source

```
<xs:element name="idMsg" type="idType" minOccurs="0">
  <xs:annotation>
    <xs:documentation>Reference to the message that this message answers.</xs:documentation>
  </xs:annotation>
</xs:element>
```

element **referencesTo/idVehicle**

diagram



type **idType**

properties

isRef	0
minOcc	0
maxOcc	1
content	complex

attributes

Name	Type	Use	Default	Fixed	annotation
src	**xs:string**	required			
id	**xs:string**	required			
unique	**xs:boolean**	optional	false		

annotation

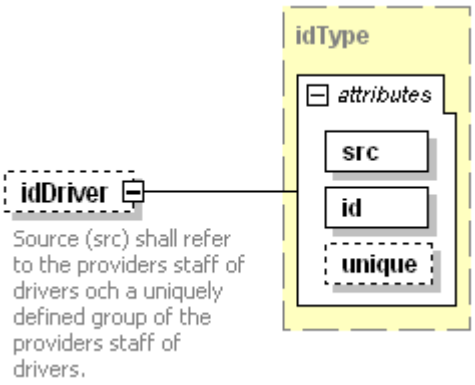
documentation

Source (src) shall refer to the whole fleet of the provider och a uniquely defined group of the providers fleet.

source
```
<xs:element name="idVehicle" type="idType" minOccurs="0">
  <xs:annotation>
    <xs:documentation>Source (src) shall refer to the whole fleet of the provider och a uniquely defined group of the providers fleet.</xs:documentation>
  </xs:annotation>
</xs:element>
```

element **referencesTo/idDriver**

diagram



type **idType**

properties

isRef	0
minOcc	0
maxOcc	1
content	complex

attributes

Name	Type	Use	Default	Fixed	annotation
src	**xs:string**	required			
id	**xs:string**	required			
unique	**xs:boolean**	optional	false		

annotation

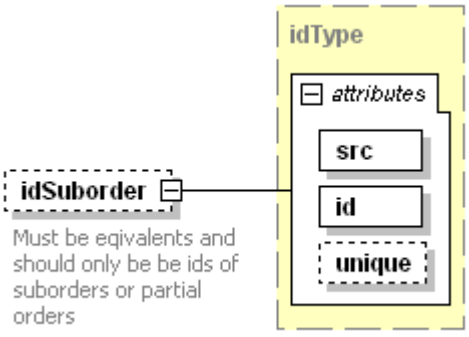
documentation

Source (src) shall refer to the providers staff of drivers och a uniquely defined group of the providers staff of drivers.

source
```
<xs:element name="idDriver" type="idType" minOccurs="0">
  <xs:annotation>
    <xs:documentation>Source (src) shall refer to the providers staff of drivers och a uniquely defined group of the providers staff of drivers.</xs:documentation>
  </xs:annotation>
</xs:element>
```

element **referencesTo/idSuborder**

diagram



type **idType**

properties

isRef	0
minOcc	0
maxOcc	1

content complex

attributes

Name	Type	Use	Default	Fixed	annotation
src	**xs:string**	required			
id	**xs:string**	required			
unique	**xs:boolean**	optional	false		

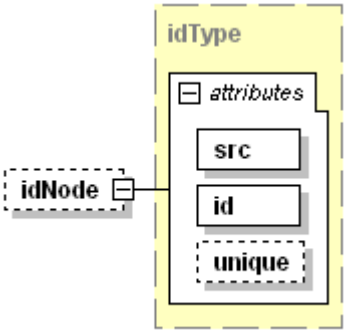
annotation
documentation
Must be eqivalents and should only be be ids of suborders or partial orders

source

```
<xs:element name="idSuborder" type="idType" minOccurs="0">
 <xs:annotation>
  <xs:documentation>Must be eqivalents and should only be be ids of suborders or partial
orders</xs:documentation>
 </xs:annotation>
</xs:element>
```

element **referencesTo/idNode**

diagram



type **idType**

properties
isRef 0
minOcc 0
maxOcc 1
content complex

attributes

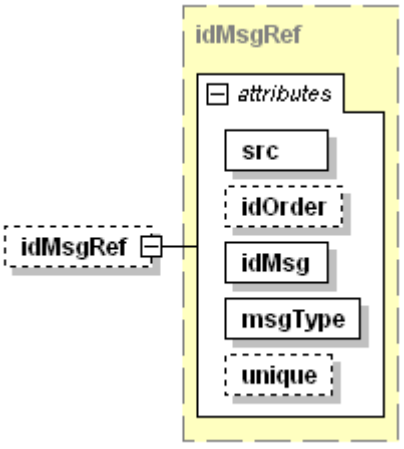
Name	Type	Use	Default	Fixed	annotation
src	**xs:string**	required			
id	**xs:string**	required			
unique	**xs:boolean**	optional	false		

source

```
<xs:element name="idNode" type="idType" minOccurs="0"/>
```

element **referencesTo/idMsgRef**

diagram



type **idMsgRef**

properties
isRef 0
minOcc 0
maxOcc 1
content complex

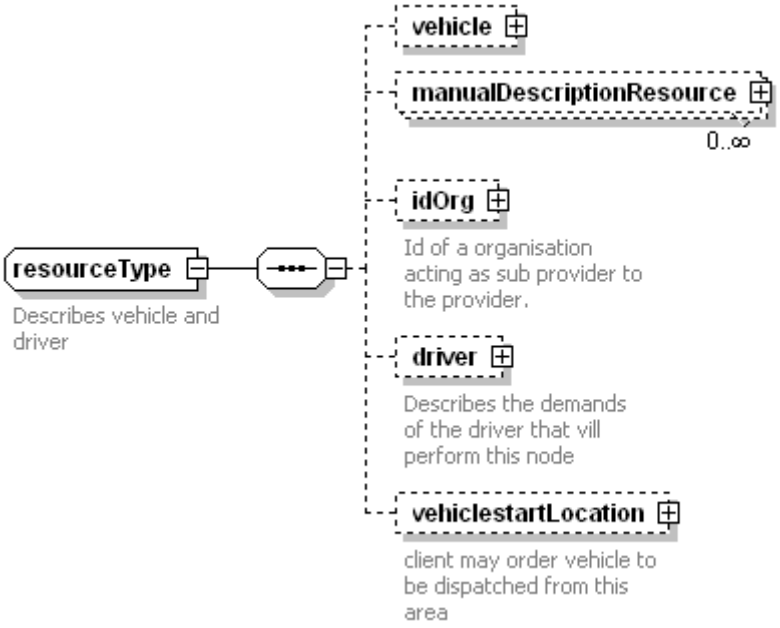
attributes

Name	Type	Use	Default	Fixed	annotation
src	**xs:string**	required			
idOrder	**xs:string**				
idMsg	**xs:string**	required			
msgType	**xs:string**	required			
unique	**xs:boolean**	optional	true		

source
```
<xs:element name="idMsgRef" type="idMsgRef " minOccurs="0"/>
```

complexType **resourceType**

diagram



children **vehicle manualDescriptionResource idOrg driver vehiclestartLocation**

used by
elements **content/resourceContent msg/resourceDispatch order/resourceOrder orderReject/resourceReject**

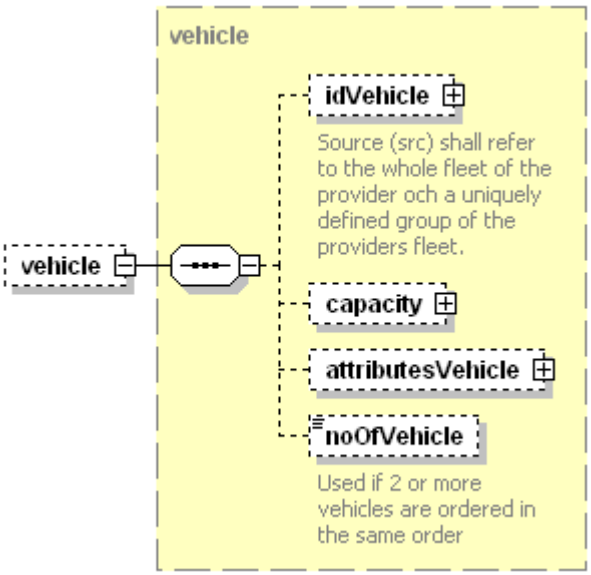
annotation
documentation
Describes vehicle and driver

source
```
<xs:complexType name="resourceType">
 <xs:annotation>
  <xs:documentation>Describes vehicle and driver</xs:documentation>
 </xs:annotation>
 <xs:sequence>
  <xs:element name="vehicle" type="vehicle" minOccurs="0"/>
  <xs:element name="manualDescriptionResource" type="manualDescriptionType" minOccurs="0"
maxOccurs="unbounded"/>
  <xs:element name="idOrg" type="idType" minOccurs="0">
   <xs:annotation>
    <xs:documentation>Id of a organisation acting as sub provider to the provider. </xs:documentation>
   </xs:annotation>
  </xs:element>
  <xs:element name="driver" type="driver" minOccurs="0">
   <xs:annotation>
    <xs:documentation>Describes the demands of the driver that vill perform this
node</xs:documentation>
   </xs:annotation>
  </xs:element>
  <xs:element name="vehiclestartLocation" type="geographicLocation" minOccurs="0">
   <xs:annotation>
    <xs:documentation>client may order vehicle to be dispatched from this area</xs:documentation>
```

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

element **resourceType/vehicle**

diagram



type **vehicle**

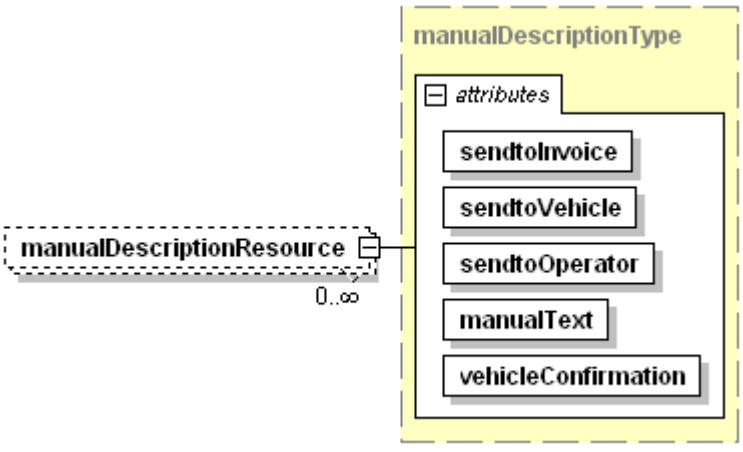
properties
- isRef 0
- minOcc 0
- maxOcc 1
- content complex

children **idVehicle** **capacity** **attributesVehicle** **noOfVehicle**

source
```
<xs:element name="vehicle" type="vehicle" minOccurs="0"/>
```

element **resourceType/manualDescriptionResource**

diagram



type **manualDescriptionType**

properties
- isRef 0
- minOcc 0
- maxOcc unbounded
- content complex

attributes

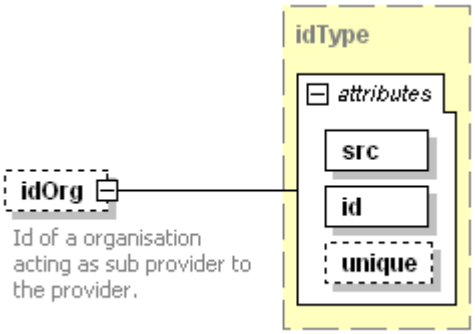
Name	Type	Use	Default	Fixed	annotation
sendtoInvoice	**xs:boolean**	required			
sendtoVehicle	**xs:boolean**	required			
sendtoOperator	**xs:boolean**	required			
manualText	**xs:string**	required			
vehicleConfirmation	**xs:boolean**	required			

source

```
<xs:element name="manualDescriptionResource" type="manualDescriptionType" minOccurs="0" maxOccurs="unbounded"/>
```

element **resourceType/idOrg**

diagram



type **idType**

properties

isRef	0
minOcc	0
maxOcc	1
content	complex

attributes

Name	Type	Use	Default	Fixed	annotation
src	**xs:string**	required			
id	**xs:string**	required			
unique	**xs:boolean**	optional	false		

annotation

documentation

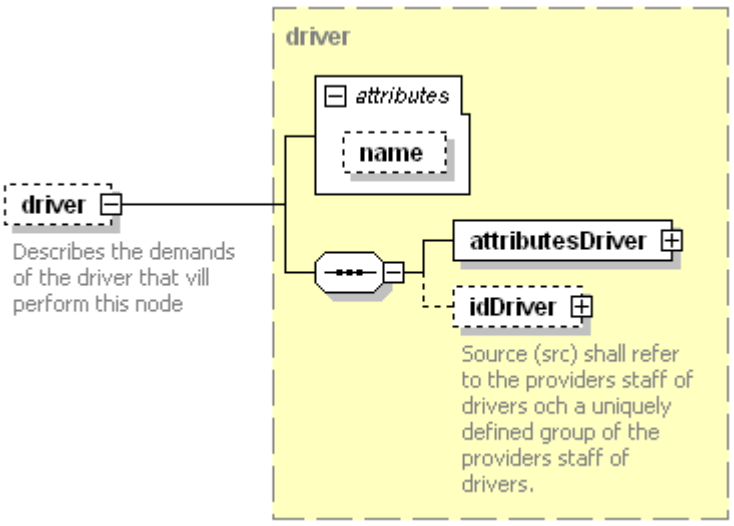
Id of a organisation acting as sub provider to the provider.

source

```
<xs:element name="idOrg" type="idType" minOccurs="0">
 <xs:annotation>
  <xs:documentation>Id of a organisation acting as sub provider to the provider. </xs:documentation>
 </xs:annotation>
</xs:element>
```

element **resourceType/driver**

diagram



type **driver**

properties
- isRef 0
- minOcc 0
- maxOcc 1
- content complex

children **attributesDriver** **idDriver**

attributes

Name	Type	Use	Default	Fixed	annotation
name	**xs:string**	optional			

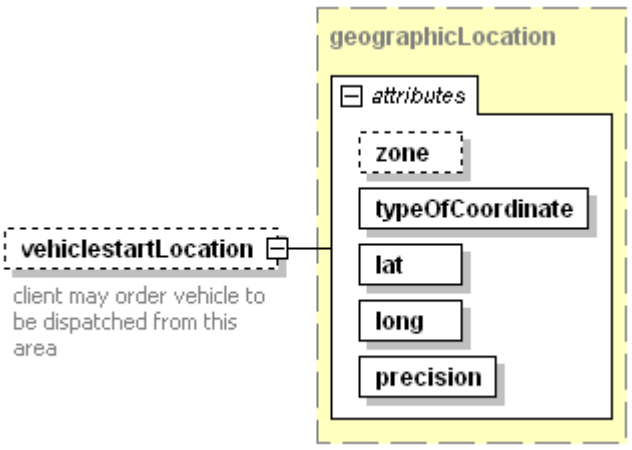
annotation
documentation
Describes the demands of the driver that vill perform this node

source

```
<xs:element name="driver" type="driver" minOccurs="0">
 <xs:annotation>
  <xs:documentation>Describes the demands of the driver that vill perform this node</xs:documentation>
 </xs:annotation>
</xs:element>
```

element **resourceType/vehiclestartLocation**

diagram



type **geographicLocation**

properties
- isRef 0
- minOcc 0
- maxOcc 1

content complex

attributes

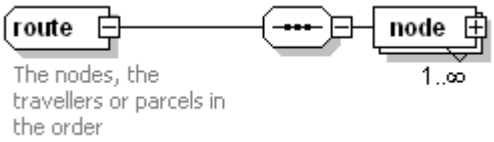
Name	Type	Use	Default	Fixed	annotation
zone	**xs:string**	optional			
typeOfCoordinate	**xs:string**	required			
lat	**xs:float**	required			
long	**xs:float**	required			
precision	**xs:integer**	required			

annotation
documentation
client may order vehicle to be dispatched from this area

source
```
<xs:element name="vehiclestartLocation" type="geographicLocation" minOccurs="0">
 <xs:annotation>
  <xs:documentation>client may order vehicle to be dispatched from this area</xs:documentation>
 </xs:annotation>
</xs:element>
```

complexType **route**

diagram



children **node**

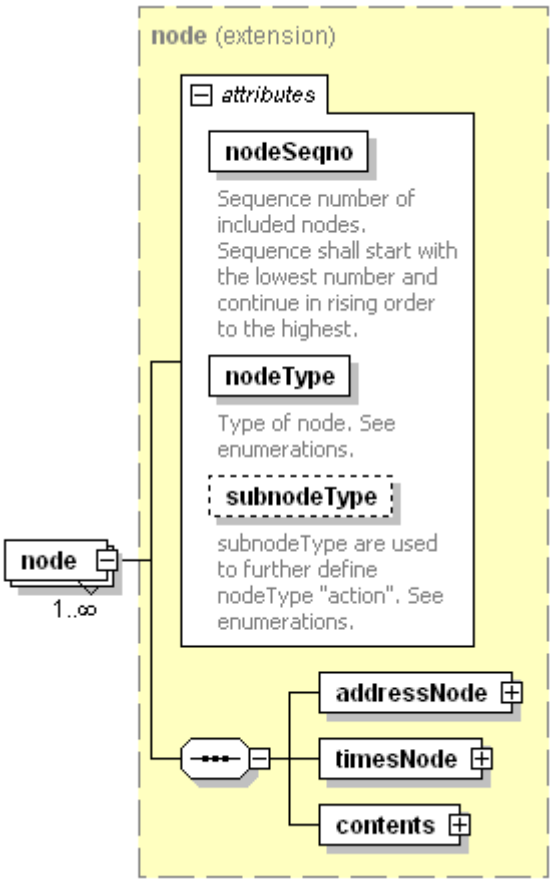
used by
element **order/route**

annotation
documentation
The nodes, the travellers or parcels in the order

source
```
<xs:complexType name="route">
 <xs:annotation>
  <xs:documentation>The nodes, the travellers or parcels in the order</xs:documentation>
 </xs:annotation>
 <xs:sequence>
  <xs:element name="node" maxOccurs="unbounded">
   <xs:complexType>
    <xs:complexContent>
     <xs:extension base="node"/>
    </xs:complexContent>
   </xs:complexType>
  </xs:element>
 </xs:sequence>
</xs:complexType>
```

element **route/node**

diagram



type extension of **node**

properties

isRef	0
minOcc	1
maxOcc	unbounded
content	complex

children **addressNode** **timesNode** **contents**

attributes

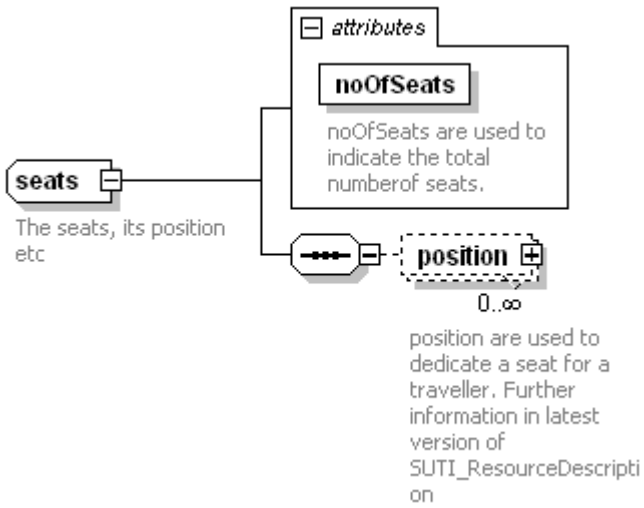
Name	Type	Use	Default	Fixed	annotation
nodeSeqno	**xs:positiveInteger**	required			documentation Sequence number of included nodes. Sequence shall start with the lowest number and continue in rising order to the highest.
nodeType	**derived by: xs:string**	required			documentation Type of node. See enumerations.
subnodeType	**derived by: xs:string**	optional			documentation subnodeType are used to

further define nodeType "action". See enumerations.

source
```
<xs:element name="node" maxOccurs="unbounded">
 <xs:complexType>
  <xs:complexContent>
   <xs:extension base="node"/>
  </xs:complexContent>
 </xs:complexType>
</xs:element>
```

complexType **seats**

diagram



children **position**

used by
element **capacity/seats**

attributes

Name	Type	Use	Default	Fixed	annotation
noOfSeats	**xs:nonNegativeInteger**	required			documentation noOfSeats are used to indicate the total numberof seats.

annotation
documentation
The seats, its position etc

source
```
<xs:complexType name="seats">
 <xs:annotation>
  <xs:documentation>The seats, its position etc</xs:documentation>
 </xs:annotation>
 <xs:sequence>
  <xs:element name="position" type="position" minOccurs="0" maxOccurs="unbounded">
   <xs:annotation>
    <xs:documentation>position are used to dedicate a seat for a traveller. Further information in latest version of SUTI_ResourceDescription</xs:documentation>
   </xs:annotation>
  </xs:element>
 </xs:sequence>
 <xs:attribute name="noOfSeats" type="xs:nonNegativeInteger" use="required">
  <xs:annotation>
   <xs:documentation>noOfSeats are used to indicate the total numberof seats.</xs:documentation>
  </xs:annotation>
 </xs:attribute>
</xs:complexType>
```

attribute **seats/@noOfSeats**

type **xs:nonNegativeInteger**

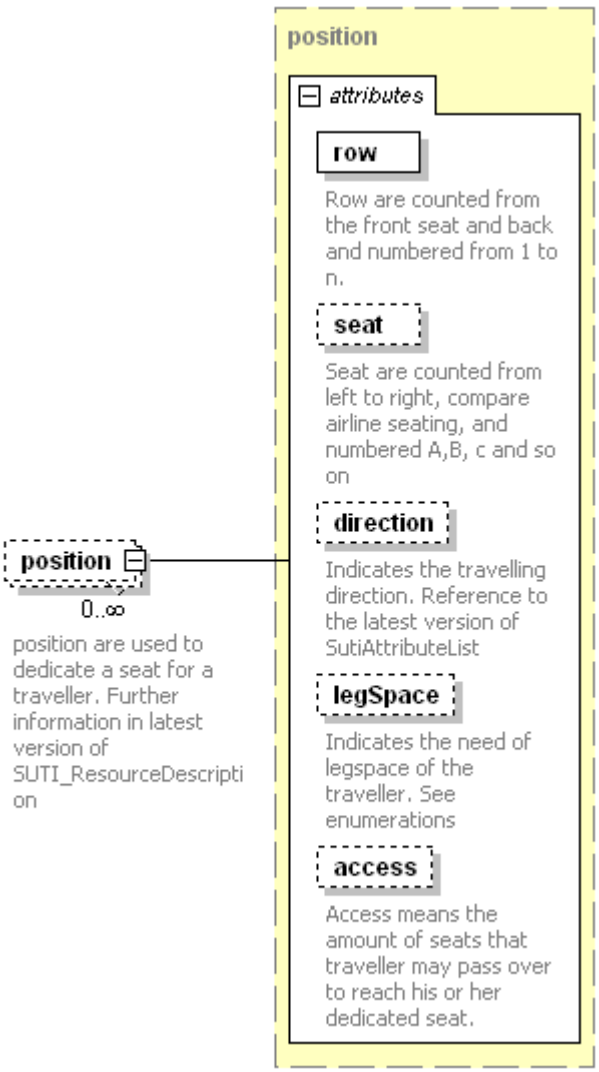
properties
- isRef 0
- use required

annotation
documentation
noOfSeats are used to indicate the total numberof seats.

source
```
<xs:attribute name="noOfSeats" type="xs:nonNegativeInteger" use="required">
 <xs:annotation>
  <xs:documentation>noOfSeats are used to indicate the total numberof seats.</xs:documentation>
 </xs:annotation>
</xs:attribute>
```

element **seats/position**

diagram



type **position**

properties
- isRef 0
- minOcc 0
- maxOcc unbounded
- content complex

attributes

Name	Type	Use	Default	Fixed	annotation
row	**xs:positiveInt**	required			documentatio

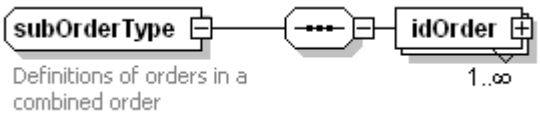
		eger		n Row are counted from the front seat and back and numbered from 1 to n.
	seat	**xs:string**	optional	documentation Seat are counted from left to right, compare airline seating, and numbered A,B, c and so on
	direction	**derived by: xs:string**	optional	documentation Indicates the travelling direction. Reference to the latest version of SutiAttributeList
	legSpace	**derived by: xs:string**	optional	documentation Indicates the need of legspace of the traveller. See enumerations
	access	**xs:nonNegativeInteger**	optional	documentation Access means the amount of seats that traveller may pass over to reach his or her dedicated seat.

annotation documentation
position are used to dedicate a seat for a traveller. Further information in latest version of SUTI_ResourceDescription

source
```
<xs:element name="position" type="position" minOccurs="0" maxOccurs="unbounded">
  <xs:annotation>
    <xs:documentation>position are used to dedicate a seat for a traveller. Further information in latest version of SUTI_ResourceDescription</xs:documentation>
  </xs:annotation>
</xs:element>
```

complexType **subOrderType**

diagram



children **idOrder**

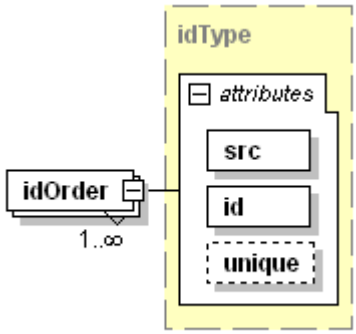
used by elements **content/subOrderContent** **msg/orderReport/economyReport/subOrderEconomy** **msg/orderReport/eventReport/event/subOrderEvent** **msg/orderLink/subOrderLink** **msg/orderReport/summaryReport/subOrderSummary**

annotation documentation
Definitions of orders in a combined order

source
```
<xs:complexType name="subOrderType">
 <xs:annotation>
  <xs:documentation>Definitions of orders in a combined order</xs:documentation>
 </xs:annotation>
 <xs:sequence>
  <xs:element name="idOrder" type="idType" maxOccurs="unbounded"/>
 </xs:sequence>
</xs:complexType>
```

element **subOrderType/idOrder**

diagram



type **idType**

properties
isRef 0
minOcc 1
maxOcc unbounded
content complex

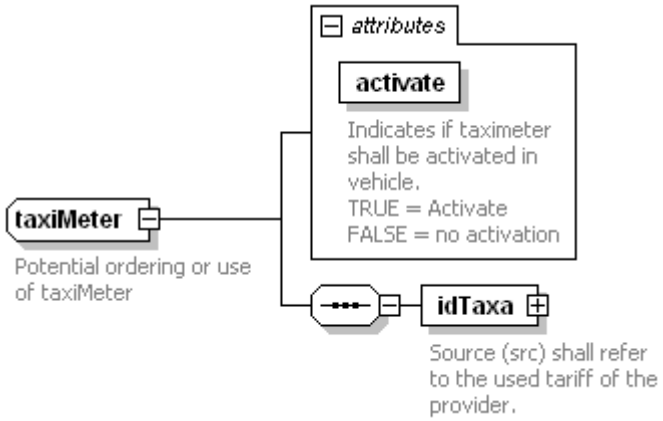
attributes

Name	Type	Use	Default	Fixed	annotation
src	**xs:string**	required			
id	**xs:string**	required			
unique	**xs:boolean**	optional	false		

source
```
<xs:element name="idOrder" type="idType" maxOccurs="unbounded"/>
```

complexType **taxiMeter**

diagram



children **idTaxa**

used by element **price/taxiMeter**

attributes	Name	Type	Use	Default	Fixed	annotation
	activate	**xs:boolean**	required			documentation Indicates if taximeter shall be activated in vehicle. TRUE = Activate FALSE = no activation

annotation
documentation
Potential ordering or use of taxiMeter

source

```
<xs:complexType name="taxiMeter">
 <xs:annotation>
  <xs:documentation>Potential ordering or use of taxiMeter</xs:documentation>
 </xs:annotation>
 <xs:sequence>
  <xs:element name="idTaxa" type="idType">
   <xs:annotation>
    <xs:documentation>Source (src) shall refer to the used tariff of the provider.</xs:documentation>
   </xs:annotation>
  </xs:element>
 </xs:sequence>
 <xs:attribute name="activate" type="xs:boolean" use="required">
  <xs:annotation>
   <xs:documentation>Indicates if taximeter shall be activated in vehicle.
TRUE = Activate
FALSE = no activation</xs:documentation>
  </xs:annotation>
 </xs:attribute>
</xs:complexType>
```

attribute **taxiMeter/@activate**

type **xs:boolean**

properties
isRef 0
use required

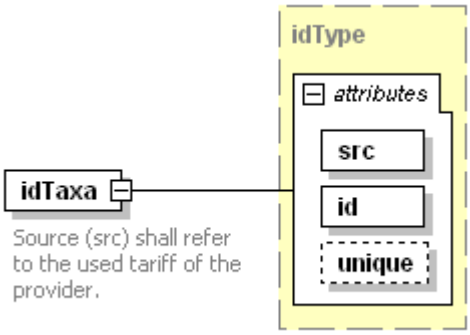
annotation
documentation
Indicates if taximeter shall be activated in vehicle.
TRUE = Activate
FALSE = no activation

source

```
<xs:attribute name="activate" type="xs:boolean" use="required">
 <xs:annotation>
  <xs:documentation>Indicates if taximeter shall be activated in vehicle.
TRUE = Activate
FALSE = no activation</xs:documentation>
 </xs:annotation>
</xs:attribute>
```

element **taxiMeter/idTaxa**

diagram



type **idType**

properties
- isRef 0
- content complex

attributes

Name	Type	Use	Default	Fixed	annotation
src	**xs:string**	required			
id	**xs:string**	required			
unique	**xs:boolean**	optional	false		

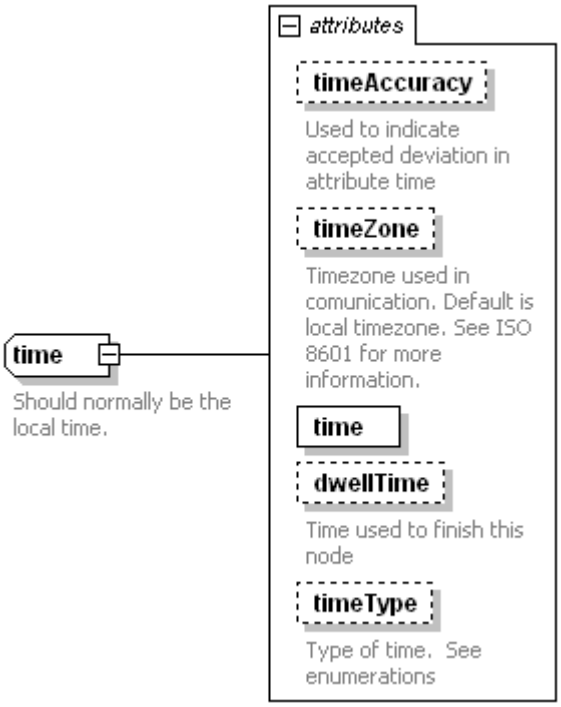
annotation
documentation
Source (src) shall refer to the used tariff of the provider.

source

```
<xs:element name="idTaxa" type="idType">
 <xs:annotation>
  <xs:documentation>Source (src) shall refer to the used tariff of the provider.</xs:documentation>
 </xs:annotation>
</xs:element>
```

complexType **time**

diagram



used by
- elements **msg/orderReport/eventReport/event/eventTime** **timesType/time** **msg/locationRequest/timeFrom** **msg/locationRequest/timeTo**

attributes

Name	Type	Use	Default	Fixed	annotation
timeAccuracy	**xs:string**	optional			documentatio

			n Used to indicate accepted deviation in attribute time
timeZone	**xs:integer**	optional	documentation Timezone used in comunication. Default is local timezone. See ISO 8601 for more information.
time	**xs:dateTime**	required	
dwellTime	**xs:int**	optional	documentation Time used to finish this node
timeType	**derived by: xs:string**	optional	documentation Type of time. See enumerations

annotation documentation
Should normally be the local time.

source
```
<xs:complexType name="time">
 <xs:annotation>
  <xs:documentation>Should normally be the local time.</xs:documentation>
 </xs:annotation>
 <xs:attribute name="timeAccuracy" type="xs:string" use="optional">
  <xs:annotation>
   <xs:documentation>Used to indicate accepted deviation in attribute time</xs:documentation>
  </xs:annotation>
 </xs:attribute>
 <xs:attribute name="timeZone" type="xs:integer" use="optional">
  <xs:annotation>
   <xs:documentation>Timezone used in comunication. Default is local timezone. See ISO 8601 for more
information.</xs:documentation>
  </xs:annotation>
 </xs:attribute>
 <xs:attribute name="time" type="xs:dateTime" use="required" form="unqualified"/>
 <xs:attribute name="dwellTime" type="xs:int" use="optional">
  <xs:annotation>
   <xs:documentation>Time used to finish this node</xs:documentation>
  </xs:annotation>
 </xs:attribute>
 <xs:attribute name="timeType" use="optional">
  <xs:annotation>
   <xs:documentation>Type of time.  See enumerations</xs:documentation>
  </xs:annotation>
  <xs:simpleType>
   <xs:restriction base="xs:string"/>
  </xs:simpleType>
 </xs:attribute>
</xs:complexType>
```

attribute **time/@timeAccuracy**

type **xs:string**

properties
isRef 0
use optional

annotation
documentation
Used to indicate accepted deviation in attribute time

source
```
<xs:attribute name="timeAccuracy" type="xs:string" use="optional">
 <xs:annotation>
  <xs:documentation>Used to indicate accepted deviation in attribute time</xs:documentation>
 </xs:annotation>
</xs:attribute>
```

attribute **time/@timeZone**

type **xs:integer**

properties
isRef 0
use optional

annotation
documentation
Timezone used in comunication. Default is local timezone. See ISO 8601 for more information.

source
```
<xs:attribute name="timeZone" type="xs:integer" use="optional">
 <xs:annotation>
  <xs:documentation>Timezone used in comunication. Default is local timezone. See ISO 8601 for more
information.</xs:documentation>
 </xs:annotation>
</xs:attribute>
```

attribute **time/@time**

type **xs:dateTime**

properties
isRef 0
form unqualified
use required

source
```
<xs:attribute name="time" type="xs:dateTime" use="required" form="unqualified"/>
```

attribute **time/@dwellTime**

type **xs:int**

properties
isRef 0
use optional

annotation
documentation
Time used to finish this node

source
```
<xs:attribute name="dwellTime" type="xs:int" use="optional">
 <xs:annotation>
  <xs:documentation>Time used to finish this node</xs:documentation>
 </xs:annotation>
</xs:attribute>
```

attribute **time/@timeType**

type restriction of **xs:string**

properties
isRef 0
use optional

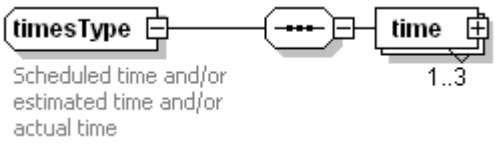
annotation
documentation
Type of time. See enumerations

source
```
<xs:attribute name="timeType" use="optional">
 <xs:annotation>
  <xs:documentation>Type of time.  See enumerations</xs:documentation>
 </xs:annotation>
 <xs:simpleType>
  <xs:restriction base="xs:string"/>
```

```
  </xs:simpleType>
</xs:attribute>
```

complexType **timesType**

diagram



children **time**

used by elements **node/timesNode** **associatedReservation/timesReservation**

annotation documentation

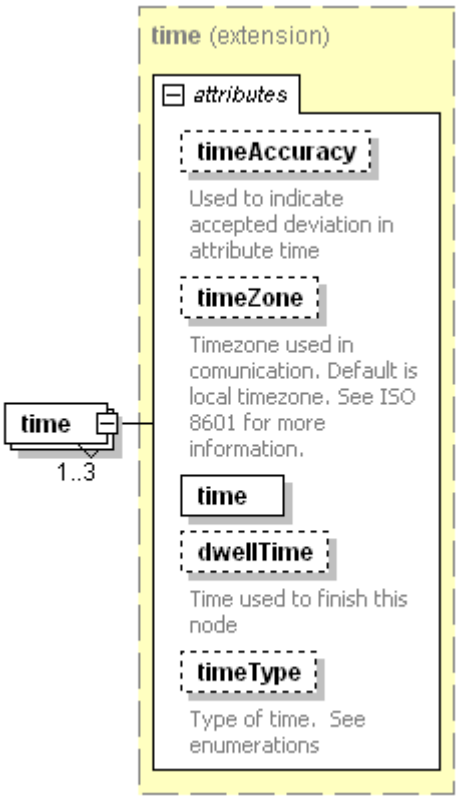
Scheduled time and/or estimated time and/or actual time

source

```
<xs:complexType name="timesType">
 <xs:annotation>
  <xs:documentation>Scheduled time and/or estimated time and/or actual time</xs:documentation>
 </xs:annotation>
 <xs:sequence>
  <xs:element name="time" maxOccurs="3">
   <xs:complexType>
    <xs:complexContent>
     <xs:extension base="time"/>
    </xs:complexContent>
   </xs:complexType>
  </xs:element>
 </xs:sequence>
</xs:complexType>
```

element **timesType/time**

diagram



type extension of **time**

properties
- isRef 0
- minOcc 1
- maxOcc 3
- content complex

attributes

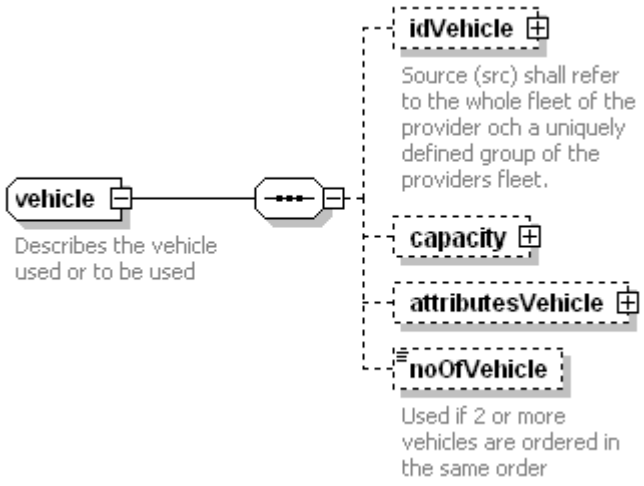
Name	Type	Use	Default	Fixed	annotation
timeAccuracy	**xs:string**	optional			documentation: Used to indicate accepted deviation in attribute time
timeZone	**xs:integer**	optional			documentation: Timezone used in comunication. Default is local timezone. See ISO 8601 for more information.
time	**xs:dateTime**	required			
dwellTime	**xs:int**	optional			documentation: Time used to finish this node
timeType	**derived by: xs:string**	optional			documentation: Type of time. See enumerations

source

```
<xs:element name="time" maxOccurs="3">
 <xs:complexType>
  <xs:complexContent>
   <xs:extension base="time"/>
  </xs:complexContent>
 </xs:complexType>
</xs:element>
```

complexType **vehicle**

diagram



children **idVehicle** **capacity** **attributesVehicle** **noOfVehicle**

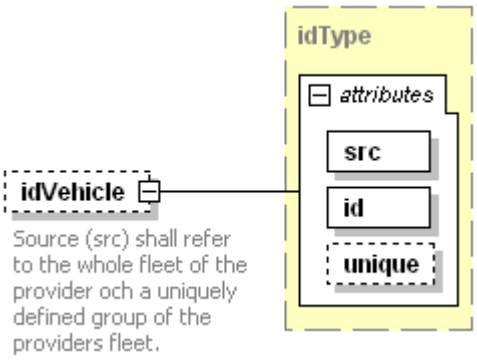
used by element **resourceType/vehicle**

annotation documentation
Describes the vehicle used or to be used

source
```
<xs:complexType name="vehicle">
 <xs:annotation>
  <xs:documentation>Describes the vehicle used or to be used</xs:documentation>
 </xs:annotation>
 <xs:sequence>
  <xs:element name="idVehicle" type="idType" minOccurs="0">
   <xs:annotation>
    <xs:documentation>Source (src) shall refer to the whole fleet of the provider och a uniquely defined group of the providers fleet.</xs:documentation>
   </xs:annotation>
  </xs:element>
  <xs:element name="capacity" type="capacity" minOccurs="0"/>
  <xs:element name="attributesVehicle" type="attributesType" minOccurs="0"/>
  <xs:element name="noOfVehicle" type="xs:positiveInteger" minOccurs="0">
   <xs:annotation>
    <xs:documentation>Used if 2 or more vehicles are ordered in the same order</xs:documentation>
   </xs:annotation>
  </xs:element>
 </xs:sequence>
</xs:complexType>
```

element **vehicle/idVehicle**

diagram



type **idType**

properties
- isRef 0
- minOcc 0
- maxOcc 1
- content complex

attributes

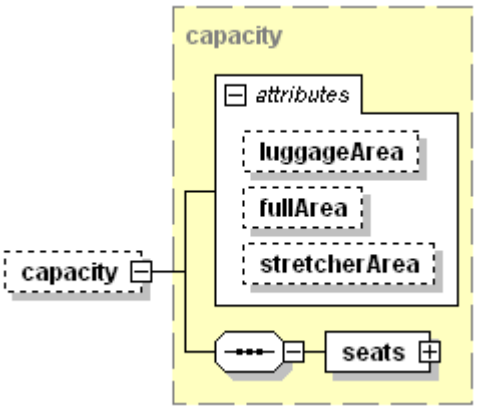
Name	Type	Use	Default	Fixed	annotation
src	**xs:string**	required			
id	**xs:string**	required			
unique	**xs:boolean**	optional	false		

annotation documentation
Source (src) shall refer to the whole fleet of the provider och a uniquely defined group of the providers fleet.

source
```
<xs:element name="idVehicle" type="idType" minOccurs="0">
 <xs:annotation>
  <xs:documentation>Source (src) shall refer to the whole fleet of the provider och a uniquely defined group of the providers fleet.</xs:documentation>
 </xs:annotation>
</xs:element>
```

element **vehicle/capacity**

diagram



type **capacity**

properties
- isRef 0
- minOcc 0
- maxOcc 1
- content complex

children **seats**

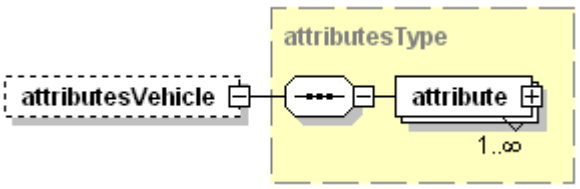
attributes

Name	Type	Use	Default	Fixed	annotation
luggageArea	**xs:float**	optional			
fullArea	**xs:float**	optional			
stretcherArea	**xs:float**	optional			

source
```
<xs:element name="capacity" type="capacity" minOccurs="0"/>
```

element **vehicle/attributesVehicle**

diagram



type **attributesType**

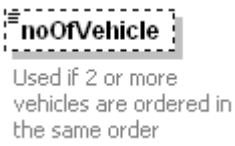
properties
- isRef 0
- minOcc 0
- maxOcc 1
- content complex

children **attribute**

source
```
<xs:element name="attributesVehicle" type="attributesType" minOccurs="0"/>
```

element **vehicle/noOfVehicle**

diagram



type **xs:positiveInteger**

properties
- isRef 0
- minOcc 0

maxOcc 1
content simple
annotation documentation
Used if 2 or more vehicles are ordered in the same order
source

```
<xs:element name="noOfVehicle" type="xs:positiveInteger" minOccurs="0">
 <xs:annotation>
  <xs:documentation>Used if 2 or more vehicles are ordered in the same order</xs:documentation>
 </xs:annotation>
</xs:element>
```