

Realis ITS

Version 29.10.2020

# DatexII 2.3 profile realisVmsStatus-1.0

# DatexII 2.3 Profile realisVmsStatus-1.0

---

## Table of Contents

- [Schema Document Properties](#)
- [Global Declarations](#)
  - [Element: d2LogicalModel](#)
- [Global Definitions](#)
  - [Complex Type: AffectedCarriagewayAndLanes](#)
  - [Complex Type: AlertCDirection](#)
  - [Complex Type: AlertCLinear](#)
  - [Complex Type: AlertCLinearByCode](#)
  - [Complex Type: AlertCLocation](#)
  - [Complex Type: AlertCMethod2Linear](#)
  - [Complex Type: AlertCMethod2Point](#)
  - [Complex Type: AlertCMethod2PrimaryPointLocation](#)
  - [Complex Type: AlertCMethod2SecondaryPointLocation](#)
  - [Complex Type: AlertCMethod4Linear](#)
  - [Complex Type: AlertCMethod4Point](#)
  - [Complex Type: AlertCMethod4PrimaryPointLocation](#)
  - [Complex Type: AlertCMethod4SecondaryPointLocation](#)
  - [Complex Type: AlertCPoint](#)
  - [Complex Type: AxleFlowValue](#)
  - [Complex Type: ConcentrationOfVehiclesValue](#)
  - [Complex Type: D2LogicalModel](#)
  - [Complex Type: DataValue](#)
  - [Complex Type: DateTimeValue](#)
  - [Complex Type: DistanceAlongLinearElement](#)
  - [Complex Type: DistanceFromLinearElementReferent](#)
  - [Complex Type: DistanceFromLinearElementStart](#)
  - [Complex Type: DurationValue](#)
  - [Complex Type: ElaboratedDataFault](#)
  - [Complex Type: Exchange](#)
  - [Complex Type: Fault](#)
  - [Complex Type: GroupOfLocations](#)
  - [Complex Type: HeaderInformation](#)
  - [Complex Type: InternationalIdentifier](#)
  - [Complex Type: Junction](#)
  - [Complex Type: Linear](#)
  - [Complex Type: LinearElement](#)
  - [Complex Type: LinearElementByCode](#)
  - [Complex Type: LinearElementByPoints](#)
  - [Complex Type: LinearWithinLinearElement](#)
  - [Complex Type: Location](#)
  - [Complex Type: MeasurementEquipmentFault](#)
  - [Complex Type: MultilingualString](#)
  - [Complex Type: MultilingualStringValue](#)
  - [Complex Type: NetworkLocation](#)
  - [Complex Type: OccupancyChangeValue](#)
  - [Complex Type: OffsetDistance](#)
  - [Complex Type: OpenlrBaseLocationReferencePoint](#)
  - [Complex Type: OpenlrBasePointLocation](#)
  - [Complex Type: OpenlrExtendedLinear](#)
  - [Complex Type: OpenlrExtendedPoint](#)
  - [Complex Type: OpenlrGeoCoordinate](#)
  - [Complex Type: OpenlrLastLocationReferencePoint](#)
  - [Complex Type: OpenlrLineAttributes](#)
  - [Complex Type: OpenlrLineLocationReference](#)
  - [Complex Type: OpenlrLocationReferencePoint](#)
  - [Complex Type: OpenlrOffsets](#)
  - [Complex Type: OpenlrPathAttributes](#)
  - [Complex Type: OpenlrPoiWithAccessPoint](#)
  - [Complex Type: OpenlrPointAlongLine](#)
  - [Complex Type: OpenlrPointLocationReference](#)
  - [Complex Type: PayloadPublication](#)
  - [Complex Type: PcuFlowValue](#)
  - [Complex Type: PercentageDistanceAlongLinearElement](#)
  - [Complex Type: PictogramDisplayAreaSettings](#)
  - [Complex Type: Point](#)
  - [Complex Type: PointAlongLinearElement](#)
  - [Complex Type: PointByCoordinates](#)
  - [Complex Type: PointCoordinates](#)
  - [Complex Type: PointExtended](#)
  - [Complex Type: Referent](#)
  - [Complex Type: Road](#)
  - [Complex Type: RoadNode](#)
  - [Complex Type: SupplementaryPositionalDescription](#)
  - [Complex Type: TextDisplayAreaSettings](#)
  - [Complex Type: TpegDescriptor](#)
  - [Complex Type: TpegFramedPoint](#)
  - [Complex Type: TpegIlocPointDescriptor](#)
  - [Complex Type: TpegJunction](#)
  - [Complex Type: TpegJunctionPointDescriptor](#)
  - [Complex Type: TpegLinearLocation](#)
  - [Complex Type: TpegNonJunctionPoint](#)
  - [Complex Type: TpegOtherPointDescriptor](#)
  - [Complex Type: TpegPoint](#)
  - [Complex Type: TpegPointDescriptor](#)
  - [Complex Type: TpegPointLocation](#)
  - [Complex Type: TpegSimplePoint](#)
  - [Complex Type: TrafficStatusValue](#)
  - [Complex Type: VehicleCountValue](#)
  - [Complex Type: VehicleFlowValue](#)
  - [Complex Type: VersionedReference](#)
  - [Complex Type: Vms](#)
  - [Complex Type: VmsFault](#)
  - [Complex Type: VmsManagedLogicalLocation](#)
  - [Complex Type: VmsMessage](#)
  - [Complex Type: VmsPictogram](#)
  - [Complex Type: VmsPictogramDisplayArea](#)
  - [Complex Type: VmsPublication](#)

- [Complex Type: VmsSetting](#)
- [Complex Type: VmsSupplementaryPanel](#)
- [Complex Type: VmsSupplementaryPictogram](#)
- [Complex Type: VmsText](#)
- [Complex Type: VmsTextLine](#)
- [Complex Type: VmsUnit](#)
- [Complex Type: VmsUnitFault](#)
- [Complex Type: ExtensionType](#)
- [Complex Type: IntermediatePointOnLinearElement](#)
- [Complex Type: LinearExtensionType](#)
- [Complex Type: PointExtensionType](#)
- [Complex Type: TextPage](#)
- [Complex Type: VmsMessageIndexVmsMessage](#)
- [Complex Type: VmsMessagePictogramDisplayAreaIndexVmsPictogramDisplayArea](#)
- [Complex Type: VmsPictogramDisplayAreaIndexPictogramDisplayAreaSettings](#)
- [Complex Type: VmsPictogramDisplayAreaPictogramSequencingIndexVmsPictogram](#)
- [Complex Type: VmsTextLineIndexVmsTextLine](#)
- [Complex Type: VmsUnitRecordVersionedReference](#)
- [Complex Type: VmsUnitTableVersionedReference](#)
- [Complex Type: VmsUnitVmsIndexVms](#)
- [Simple Type: AlertCDirectionEnum](#)
- [Simple Type: AlertCLocationCode](#)
- [Simple Type: AngleInDegrees](#)
- [Simple Type: AreaOfInterestEnum](#)
- [Simple Type: AxlesPerHour](#)
- [Simple Type: Boolean](#)
- [Simple Type: CarriagewayEnum](#)
- [Simple Type: CodedReasonForSettingMessageEnum](#)
- [Simple Type: ColourEnum](#)
- [Simple Type: ComputationMethodEnum](#)
- [Simple Type: ConcentrationVehiclesPerKilometre](#)
- [Simple Type: ConfidentialityValueEnum](#)
- [Simple Type: CountryEnum](#)
- [Simple Type: DateTime](#)
- [Simple Type: DirectionEnum](#)
- [Simple Type: ElaboratedDataFaultEnum](#)
- [Simple Type: FaultSeverityEnum](#)
- [Simple Type: Float](#)
- [Simple Type: HeightGradeEnum](#)
- [Simple Type: InformationStatusEnum](#)
- [Simple Type: Integer](#)
- [Simple Type: JunctionClassificationEnum](#)
- [Simple Type: KilometresPerHour](#)
- [Simple Type: LaneEnum](#)
- [Simple Type: Language](#)
- [Simple Type: LinearElementNatureEnum](#)
- [Simple Type: LinearReferencingDirectionEnum](#)
- [Simple Type: LocationDescriptorEnum](#)
- [Simple Type: MeasurementEquipmentFaultEnum](#)
- [Simple Type: MetresAsFloat](#)
- [Simple Type: MetresAsNonNegativeInteger](#)
- [Simple Type: MultilingualStringValue](#)
- [Simple Type: NonNegativeInteger](#)
- [Simple Type: OpenIrFormOfWayEnum](#)
- [Simple Type: OpenIrFunctionalRoadClassEnum](#)
- [Simple Type: OpenIrOrientationEnum](#)
- [Simple Type: OpenIrSideOfRoadEnum](#)
- [Simple Type: PassengerCarUnitsPerHour](#)
- [Simple Type: Percentage](#)
- [Simple Type: ReferentTypeEnum](#)
- [Simple Type: RoadTypeEnum](#)
- [Simple Type: Seconds](#)
- [Simple Type: String](#)
- [Simple Type: Tonnes](#)
- [Simple Type: TpegLoc01FramedPointLocationSubtypeEnum](#)
- [Simple Type: TpegLoc01LinearLocationSubtypeEnum](#)
- [Simple Type: TpegLoc01SimplePointLocationSubtypeEnum](#)
- [Simple Type: TpegLoc03IcPointDescriptorSubtypeEnum](#)
- [Simple Type: TpegLoc03JunctionPointDescriptorSubtypeEnum](#)
- [Simple Type: TpegLoc03OtherPointDescriptorSubtypeEnum](#)
- [Simple Type: TrafficStatusEnum](#)
- [Simple Type: UrgencyEnum](#)
- [Simple Type: Url](#)
- [Simple Type: VehiclesPerHour](#)
- [Simple Type: VmsDatexPictogramEnum](#)
- [Simple Type: VmsDatexSupplementalPictogramEnum](#)
- [Simple Type: VmsFaultEnum](#)
- [Simple Type: VmsLuminanceLevelEnum](#)
- [Simple Type: VmsMessageInformationTypeEnum](#)

[top](#)

## Schema Document Properties

<b>Target Namespace</b>	<a href="http://datex2.eu/schema/2/2_0">http://datex2.eu/schema/2/2_0</a>
<b>Version</b>	2.3
<b>Element and Attribute Namespaces</b>	

- Global element and attribute declarations belong to this schema's target namespace.
- By default, local element declarations belong to this schema's target namespace.
- By default, local attribute declarations have no namespace.

## Declared Namespaces

Prefix	Namespace
xml	<a href="http://www.w3.org/XML/1998/namespace">http://www.w3.org/XML/1998/namespace</a>
xs	<a href="http://www.w3.org/2001/XMLSchema">http://www.w3.org/2001/XMLSchema</a>
D2LogicalModel	<a href="http://datex2.eu/schema/2/2_0">http://datex2.eu/schema/2/2_0</a>

## Schema Component Representation

```
<xs:schema elementFormDefault="qualified" attributeFormDefault="unqualified" version="2.3"
targetNamespace="http://datex2.eu/schema/2/2_0">
...
</xs:schema>
```

[top](#)

## Global Declarations

### Element: **d2LogicalModel**

<b>Name</b>	d2LogicalModel
<b>Type</b>	<a href="#">D2LogicalModel:D2LogicalModel</a>
<b>Nilable</b>	no
<b>Abstract</b>	no

#### XML Instance Representation

```
<D2LogicalModel:d2LogicalModel
modelBaseVersion="2 [1]">
  <D2LogicalModel:exchange> D2LogicalModel:Exchange </D2LogicalModel:exchange> [1]
  <D2LogicalModel:payloadPublication> D2LogicalModel:PayloadPublication </D2LogicalModel:payloadPublication> [0..1]
  <D2LogicalModel:d2LogicalModelExtension> D2LogicalModel:_ExtensionType </D2LogicalModel:d2LogicalModelExtension>
  [0..1]
</D2LogicalModel:d2LogicalModel>
```

#### Schema Component Representation

```
<xs:element name="d2LogicalModel" type="D2LogicalModel:D2LogicalModel"/>
```

[top](#)

## Global Definitions

### Complex Type: **AffectedCarriagewayAndLanes**

<b>Super-types:</b>	None
<b>Sub-types:</b>	None

<b>Name</b>	AffectedCarriagewayAndLanes
<b>Abstract</b>	no
<b>Documentation</b>	Supplementary positional information which details carriageway and lane locations. Several instances may exist where the element being described extends over more than one carriageway.

#### XML Instance Representation

```
<...>
  <D2LogicalModel:carriageway> D2LogicalModel:CarriagewayEnum </D2LogicalModel:carriageway> [1] ?
  <D2LogicalModel:lane> D2LogicalModel:LaneEnum </D2LogicalModel:lane> [0..*] ?
  <D2LogicalModel:footpath> D2LogicalModel:Boolean </D2LogicalModel:footpath> [0..1] ?
  <D2LogicalModel:lengthAffected> D2LogicalModel:MetresAsFloat </D2LogicalModel:lengthAffected> [0..1] ?
  <D2LogicalModel:affectedCarriagewayAndLanesExtension> D2LogicalModel:_ExtensionType
  </D2LogicalModel:affectedCarriagewayAndLanesExtension> [0..1]
</...>
```

#### Schema Component Representation

```
<xs:complexType name="AffectedCarriagewayAndLanes">
  <xs:sequence>
    <xs:element name="carriageway" type="D2LogicalModel:CarriagewayEnum" minOccurs="1" maxOccurs="1"/>
    <xs:element name="lane" type="D2LogicalModel:LaneEnum" minOccurs="0" maxOccurs="unbounded"/>
    <xs:element name="footpath" type="D2LogicalModel:Boolean" minOccurs="0" maxOccurs="1"/>
    <xs:element name="lengthAffected" type="D2LogicalModel:MetresAsFloat" minOccurs="0" maxOccurs="1"/>
    <xs:element name="affectedCarriagewayAndLanesExtension" type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
```

[top](#)

### Complex Type: **AlertCDirection**

<b>Super-types:</b>	None
<b>Sub-types:</b>	None

<b>Name</b>	AlertCDirection
<b>Abstract</b>	no
<b>Documentation</b>	The direction of traffic flow along the road to which the information relates.

#### XML Instance Representation

```
<...>
  <D2LogicalModel:alertCDirectionCoded> D2LogicalModel:AlertCDirectionEnum </D2LogicalModel:alertCDirectionCoded>
  [1] ?
  <D2LogicalModel:alertCDirectionNamed> D2LogicalModel:MultilingualString </D2LogicalModel:alertCDirectionNamed>
  [0..1] ?
  <D2LogicalModel:alertCDirectionSense> D2LogicalModel:Boolean </D2LogicalModel:alertCDirectionSense> [0..1] ?
  <D2LogicalModel:alertCDirectionExtension> D2LogicalModel:_ExtensionType </D2LogicalModel:alertCDirectionExtension>
  [0..1]
</...>
```

## Schema Component Representation

```
<xs:complexType name="AlertCDirection">
  <xs:sequence>
    <xs:element name="alertCDirectionCoded" type="D2LogicalModel:AlertCDirectionEnum" minOccurs="1" maxOccurs="1"/>
    <xs:element name="alertCDirectionNamed" type="D2LogicalModel:MultilingualString" minOccurs="0" maxOccurs="1"/>
    <xs:element name="alertCDirectionSense" type="D2LogicalModel:Boolean" minOccurs="0" maxOccurs="1"/>
    <xs:element name="alertCDirectionExtension" type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
```

[top](#)

## Complex Type: AlertCLinear

Super-types:	None
Sub-types:	<ul style="list-style-type: none"><li>• <a href="#">AlertCLinearByCode</a> (by extension)</li><li>• <a href="#">AlertCMethod2Linear</a> (by extension)</li><li>• <a href="#">AlertCMethod4Linear</a> (by extension)</li></ul>

<b>Name</b>	AlertCLinear
<b>Abstract</b>	yes
<b>Documentation</b>	A linear section along a road defined between two points on the road by reference to a pre-defined ALERT-C location table.

## XML Instance Representation

```
<...>
  <D2LogicalModel:alertCLocationCountryCode> D2LogicalModel:String </D2LogicalModel:alertCLocationCountryCode> [1] ?
  <D2LogicalModel:alertCLocationTableNumber> D2LogicalModel:String </D2LogicalModel:alertCLocationTableNumber> [1] ?
  <D2LogicalModel:alertCLocationTableVersion> D2LogicalModel:String </D2LogicalModel:alertCLocationTableVersion> [1] ?
  <D2LogicalModel:alertCLinearExtension> D2LogicalModel:_ExtensionType </D2LogicalModel:alertCLinearExtension> [0..1]
</...>
```

## Schema Component Representation

```
<xs:complexType name="AlertCLinear" abstract="true">
  <xs:sequence>
    <xs:element name="alertCLocationCountryCode" type="D2LogicalModel:String" minOccurs="1" maxOccurs="1"/>
    <xs:element name="alertCLocationTableNumber" type="D2LogicalModel:String" minOccurs="1" maxOccurs="1"/>
    <xs:element name="alertCLocationTableVersion" type="D2LogicalModel:String" minOccurs="1" maxOccurs="1"/>
    <xs:element name="alertCLinearExtension" type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
```

[top](#)

## Complex Type: AlertCLinearByCode

Super-types:	<a href="#">AlertCLinear</a> < <a href="#">AlertCLinearByCode</a> (by extension)
Sub-types:	None

<b>Name</b>	AlertCLinearByCode
<b>Abstract</b>	no
<b>Documentation</b>	A linear section along a road defined by reference to a linear section in a pre-defined ALERT-C location table.

## XML Instance Representation

```
<...>
  <D2LogicalModel:alertCLocationCountryCode> D2LogicalModel:String </D2LogicalModel:alertCLocationCountryCode> [1] ?
  <D2LogicalModel:alertCLocationTableNumber> D2LogicalModel:String </D2LogicalModel:alertCLocationTableNumber> [1] ?
  <D2LogicalModel:alertCLocationTableVersion> D2LogicalModel:String </D2LogicalModel:alertCLocationTableVersion> [1] ?
  <D2LogicalModel:alertCLinearExtension> D2LogicalModel:_ExtensionType </D2LogicalModel:alertCLinearExtension> [0..1]
  <D2LogicalModel:alertCDirection> D2LogicalModel:AlertCDirection </D2LogicalModel:alertCDirection> [1]
  <D2LogicalModel:locationCodeForLinearLocation> D2LogicalModel:AlertCLocation </D2LogicalModel:locationCodeForLinearLocation> [1] ?
  <D2LogicalModel:alertCLinearByCodeExtension> D2LogicalModel:_ExtensionType </D2LogicalModel:alertCLinearByCodeExtension> [0..1]
</...>
```

## Schema Component Representation

```
<xs:complexType name="AlertCLinearByCode">
  <xs:complexContent>
    <xs:extension base="D2LogicalModel:AlertCLinear">
      <xs:sequence>
        <xs:element name="alertCDirection" type="D2LogicalModel:AlertCDirection"/>
        <xs:element name="locationCodeForLinearLocation" type="D2LogicalModel:AlertCLocation"/>
        <xs:element name="alertCLinearByCodeExtension" type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

[top](#)

## Complex Type: AlertCLocation

Super-types:	None
Sub-types:	None

<b>Name</b>	AlertCLocation
<b>Abstract</b>	no
<b>Documentation</b>	Identification of a specific point, linear or area location in an ALERT-C location table.

#### XML Instance Representation

```
<...>
  <D2LogicalModel:alertCLocationName> D2LogicalModel:MultilingualString </D2LogicalModel:alertCLocationName> [0..1]
  ?
  <D2LogicalModel:specificLocation> D2LogicalModel:AlertCLocationCode </D2LogicalModel:specificLocation> [1] ?
  <D2LogicalModel:alertCLocationExtension> D2LogicalModel:_ExtensionType </D2LogicalModel:alertCLocationExtension>
  [0..1]
</...>
```

#### Schema Component Representation

```
<xs:complexType name="AlertCLocation">
  <xs:sequence>
    <xs:element name="alertCLocationName" type="D2LogicalModel:MultilingualString" minOccurs="0" maxOccurs="1"/>
    <xs:element name="specificLocation" type="D2LogicalModel:AlertCLocationCode" minOccurs="1" maxOccurs="1"/>
    <xs:element name="alertCLocationExtension" type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
```

[top](#)

### Complex Type: AlertCMethod2Linear

Super-types:	<a href="#">AlertCLinear</a> < AlertCMethod2Linear (by extension)
Sub-types:	None

<b>Name</b>	AlertCMethod2Linear
<b>Abstract</b>	no
<b>Documentation</b>	A linear section along a road between two points, Primary and Secondary, which are pre-defined in an ALERT-C location table. Direction is FROM the Secondary point TO the Primary point, i.e. the Primary point is downstream of the Secondary point.

#### XML Instance Representation

```
<...>
  <D2LogicalModel:alertCLocationCountryCode> D2LogicalModel:String </D2LogicalModel:alertCLocationCountryCode> [1] ?
  <D2LogicalModel:alertCLocationTableNumber> D2LogicalModel:String </D2LogicalModel:alertCLocationTableNumber> [1] ?
  <D2LogicalModel:alertCLocationTableVersion> D2LogicalModel:String </D2LogicalModel:alertCLocationTableVersion> [1]
  ?
  <D2LogicalModel:alertCLinearExtension> D2LogicalModel:_ExtensionType </D2LogicalModel:alertCLinearExtension>
  [0..1]
  <D2LogicalModel:alertCDirection> D2LogicalModel:AlertCDirection </D2LogicalModel:alertCDirection> [1]
  <D2LogicalModel:alertCMethod2PrimaryPointLocation> D2LogicalModel:AlertCMethod2PrimaryPointLocation
  </D2LogicalModel:alertCMethod2PrimaryPointLocation> [1]
  <D2LogicalModel:alertCMethod2SecondaryPointLocation> D2LogicalModel:AlertCMethod2SecondaryPointLocation
  </D2LogicalModel:alertCMethod2SecondaryPointLocation> [1]
  <D2LogicalModel:alertCMethod2LinearExtension> D2LogicalModel:_ExtensionType
  </D2LogicalModel:alertCMethod2LinearExtension> [0..1]
</...>
```

#### Schema Component Representation

```
<xs:complexType name="AlertCMethod2Linear">
  <xs:complexContent>
    <xs:extension base="D2LogicalModel:AlertCLinear">
      <xs:sequence>
        <xs:element name="alertCDirection" type="D2LogicalModel:AlertCDirection"/>
        <xs:element name="alertCMethod2PrimaryPointLocation"
          type="D2LogicalModel:AlertCMethod2PrimaryPointLocation"/>
        <xs:element name="alertCMethod2SecondaryPointLocation"
          type="D2LogicalModel:AlertCMethod2SecondaryPointLocation"/>
        <xs:element name="alertCMethod2LinearExtension" type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

[top](#)

### Complex Type: AlertCMethod2Point

Super-types:	<a href="#">AlertCPoint</a> < AlertCMethod2Point (by extension)
Sub-types:	None

<b>Name</b>	AlertCMethod2Point
<b>Abstract</b>	no
<b>Documentation</b>	A single point on the road network defined by reference to a point in a pre-defined ALERT-C location table and which has an associated direction of traffic flow.

#### XML Instance Representation

```
<...>
  <D2LogicalModel:alertCLocationCountryCode> D2LogicalModel:String </D2LogicalModel:alertCLocationCountryCode> [1] ?
  <D2LogicalModel:alertCLocationTableNumber> D2LogicalModel:String </D2LogicalModel:alertCLocationTableNumber> [1] ?
</...>
```

```

<D2LogicalModel:alertCLocationTableVersion> D2LogicalModel:String </D2LogicalModel:alertCLocationTableVersion> [1]
?
<D2LogicalModel:alertCPointExtension> D2LogicalModel:_ExtensionType </D2LogicalModel:alertCPointExtension> [0..1]
<D2LogicalModel:alertCDirection> D2LogicalModel:AlertCDirection </D2LogicalModel:alertCDirection> [1]
<D2LogicalModel:alertCMethod2PrimaryPointLocation> D2LogicalModel:AlertCMethod2PrimaryPointLocation
</D2LogicalModel:alertCMethod2PrimaryPointLocation> [1]
<D2LogicalModel:alertCMethod2PointExtension> D2LogicalModel:_ExtensionType
</D2LogicalModel:alertCMethod2PointExtension> [0..1]
</...>

```

#### Schema Component Representation

```

<xs:complexType name="AlertCMethod2Point">
  <xs:complexContent>
    <xs:extension base="D2LogicalModel:AlertCPoint">
      <xs:sequence>
        <xs:element name="alertCDirection" type="D2LogicalModel:AlertCDirection"/>
        <xs:element name="alertCMethod2PrimaryPointLocation"
          type="D2LogicalModel:AlertCMethod2PrimaryPointLocation"/>
        <xs:element name="alertCMethod2PointExtension" type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

```

[top](#)

### Complex Type: AlertCMethod2PrimaryPointLocation

Super-types:	None
Sub-types:	None

<b>Name</b>	AlertCMethod2PrimaryPointLocation
<b>Abstract</b>	no
<b>Documentation</b>	The point (called Primary point) which is either a single point or at the downstream end of a linear road section. The point is specified by a reference to a point in a pre-defined ALERT-C location table.

#### XML Instance Representation

```

<...>
<D2LogicalModel:alertCLocation> D2LogicalModel:AlertCLocation </D2LogicalModel:alertCLocation> [1]
<D2LogicalModel:alertCMethod2PrimaryPointLocationExtension> D2LogicalModel:_ExtensionType
</D2LogicalModel:alertCMethod2PrimaryPointLocationExtension> [0..1]
</...>

```

#### Schema Component Representation

```

<xs:complexType name="AlertCMethod2PrimaryPointLocation">
  <xs:sequence>
    <xs:element name="alertCLocation" type="D2LogicalModel:AlertCLocation"/>
    <xs:element name="alertCMethod2PrimaryPointLocationExtension" type="D2LogicalModel:_ExtensionType"
      minOccurs="0"/>
  </xs:sequence>
</xs:complexType>

```

[top](#)

### Complex Type: AlertCMethod2SecondaryPointLocation

Super-types:	None
Sub-types:	None

<b>Name</b>	AlertCMethod2SecondaryPointLocation
<b>Abstract</b>	no
<b>Documentation</b>	The point (called Secondary point) which is at the upstream end of a linear road section. The point is specified by a reference to a point in a pre-defined ALERT-C location table.

#### XML Instance Representation

```

<...>
<D2LogicalModel:alertCLocation> D2LogicalModel:AlertCLocation </D2LogicalModel:alertCLocation> [1]
<D2LogicalModel:alertCMethod2SecondaryPointLocationExtension> D2LogicalModel:_ExtensionType
</D2LogicalModel:alertCMethod2SecondaryPointLocationExtension> [0..1]
</...>

```

#### Schema Component Representation

```

<xs:complexType name="AlertCMethod2SecondaryPointLocation">
  <xs:sequence>
    <xs:element name="alertCLocation" type="D2LogicalModel:AlertCLocation"/>
    <xs:element name="alertCMethod2SecondaryPointLocationExtension" type="D2LogicalModel:_ExtensionType"
      minOccurs="0"/>
  </xs:sequence>
</xs:complexType>

```

[top](#)

### Complex Type: AlertCMethod4Linear

Super-types:	<a href="#">AlertCLinear</a> < AlertCMethod4Linear (by extension)
Sub-types:	None

<b>Name</b>	AlertCMethod4Linear
<b>Abstract</b>	no
<b>Documentation</b>	A linear section along a road between two points, Primary and Secondary, which are pre-defined ALERT-C locations plus offset distance. Direction is FROM the Secondary point TO the Primary point, i.e. the Primary point is downstream of the Secondary point.

#### XML Instance Representation

```
<...>
  <D2LogicalModel:alertCLocationCountryCode> D2LogicalModel:String </D2LogicalModel:alertCLocationCountryCode> [1] ?
  <D2LogicalModel:alertCLocationTableNumber> D2LogicalModel:String </D2LogicalModel:alertCLocationTableNumber> [1] ?
  <D2LogicalModel:alertCLocationTableVersion> D2LogicalModel:String </D2LogicalModel:alertCLocationTableVersion> [1] ?
  <D2LogicalModel:alertCLinearExtension> D2LogicalModel:_ExtensionType </D2LogicalModel:alertCLinearExtension> [0..1]
  <D2LogicalModel:alertCDirection> D2LogicalModel:AlertCDirection </D2LogicalModel:alertCDirection> [1]
  <D2LogicalModel:alertCMethod4PrimaryPointLocation> D2LogicalModel:AlertCMethod4PrimaryPointLocation </D2LogicalModel:alertCMethod4PrimaryPointLocation> [1]
  <D2LogicalModel:alertCMethod4SecondaryPointLocation> D2LogicalModel:AlertCMethod4SecondaryPointLocation </D2LogicalModel:alertCMethod4SecondaryPointLocation> [1]
  <D2LogicalModel:alertCMethod4LinearExtension> D2LogicalModel:_ExtensionType </D2LogicalModel:alertCMethod4LinearExtension> [0..1]
</...>
```

#### Schema Component Representation

```
<xs:complexType name="AlertCMethod4Linear">
  <xs:complexContent>
    <xs:extension base="D2LogicalModel:AlertCLinear">
      <xs:sequence>
        <xs:element name="alertCDirection" type="D2LogicalModel:AlertCDirection"/>
        <xs:element name="alertCMethod4PrimaryPointLocation" type="D2LogicalModel:AlertCMethod4PrimaryPointLocation"/>
        <xs:element name="alertCMethod4SecondaryPointLocation" type="D2LogicalModel:AlertCMethod4SecondaryPointLocation"/>
        <xs:element name="alertCMethod4LinearExtension" type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

[top](#)

### Complex Type: AlertCMethod4Point

Super-types:	<a href="#">AlertCPoint</a> < AlertCMethod4Point (by extension)
Sub-types:	None

<b>Name</b>	AlertCMethod4Point
<b>Abstract</b>	no
<b>Documentation</b>	A single point on the road network defined by reference to a point in a pre-defined ALERT-C location table plus an offset distance and which has an associated direction of traffic flow.

#### XML Instance Representation

```
<...>
  <D2LogicalModel:alertCLocationCountryCode> D2LogicalModel:String </D2LogicalModel:alertCLocationCountryCode> [1] ?
  <D2LogicalModel:alertCLocationTableNumber> D2LogicalModel:String </D2LogicalModel:alertCLocationTableNumber> [1] ?
  <D2LogicalModel:alertCLocationTableVersion> D2LogicalModel:String </D2LogicalModel:alertCLocationTableVersion> [1] ?
  <D2LogicalModel:alertCPointExtension> D2LogicalModel:_ExtensionType </D2LogicalModel:alertCPointExtension> [0..1]
  <D2LogicalModel:alertCDirection> D2LogicalModel:AlertCDirection </D2LogicalModel:alertCDirection> [1]
  <D2LogicalModel:alertCMethod4PrimaryPointLocation> D2LogicalModel:AlertCMethod4PrimaryPointLocation </D2LogicalModel:alertCMethod4PrimaryPointLocation> [1]
  <D2LogicalModel:alertCMethod4PointExtension> D2LogicalModel:_ExtensionType </D2LogicalModel:alertCMethod4PointExtension> [0..1]
</...>
```

#### Schema Component Representation

```
<xs:complexType name="AlertCMethod4Point">
  <xs:complexContent>
    <xs:extension base="D2LogicalModel:AlertCPoint">
      <xs:sequence>
        <xs:element name="alertCDirection" type="D2LogicalModel:AlertCDirection"/>
        <xs:element name="alertCMethod4PrimaryPointLocation" type="D2LogicalModel:AlertCMethod4PrimaryPointLocation"/>
        <xs:element name="alertCMethod4PointExtension" type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

[top](#)

### Complex Type: AlertCMethod4PrimaryPointLocation

Super-types:	None
Sub-types:	None

<b>Name</b>	AlertCMethod4PrimaryPointLocation
<b>Abstract</b>	no
<b>Documentation</b>	The point (called Primary point) which is either a single point or at the downstream end of a linear road section. The point is specified by a reference to a point in a pre-defined ALERT-C location table plus a non-



negative offset distance.

#### XML Instance Representation

```
<...>
  <D2LogicalModel:alertCLocation> D2LogicalModel:AlertCLocation </D2LogicalModel:alertCLocation> [1]
  <D2LogicalModel:offsetDistance> D2LogicalModel:OffsetDistance </D2LogicalModel:offsetDistance> [1]
  <D2LogicalModel:alertCMethod4PrimaryPointLocationExtension> D2LogicalModel:_ExtensionType
</D2LogicalModel:alertCMethod4PrimaryPointLocationExtension> [0..1]
</...>
```

#### Schema Component Representation

```
<xs:complexType name="AlertCMethod4PrimaryPointLocation">
  <xs:sequence>
    <xs:element name="alertCLocation" type="D2LogicalModel:AlertCLocation"/>
    <xs:element name="offsetDistance" type="D2LogicalModel:OffsetDistance"/>
    <xs:element name="alertCMethod4PrimaryPointLocationExtension" type="D2LogicalModel:_ExtensionType"
      minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
```

[top](#)

### Complex Type: AlertCMethod4SecondaryPointLocation

Super-types: None  
Sub-types: None

**Name** AlertCMethod4SecondaryPointLocation  
**Abstract** no  
**Documentation** The point (called Secondary point) which is at the upstream end of a linear road section. The point is specified by a reference to a point in a pre-defined Alert-C location table plus a non-negative offset distance.

#### XML Instance Representation

```
<...>
  <D2LogicalModel:alertCLocation> D2LogicalModel:AlertCLocation </D2LogicalModel:alertCLocation> [1]
  <D2LogicalModel:offsetDistance> D2LogicalModel:OffsetDistance </D2LogicalModel:offsetDistance> [1]
  <D2LogicalModel:alertCMethod4SecondaryPointLocationExtension> D2LogicalModel:_ExtensionType
</D2LogicalModel:alertCMethod4SecondaryPointLocationExtension> [0..1]
</...>
```

#### Schema Component Representation

```
<xs:complexType name="AlertCMethod4SecondaryPointLocation">
  <xs:sequence>
    <xs:element name="alertCLocation" type="D2LogicalModel:AlertCLocation"/>
    <xs:element name="offsetDistance" type="D2LogicalModel:OffsetDistance"/>
    <xs:element name="alertCMethod4SecondaryPointLocationExtension" type="D2LogicalModel:_ExtensionType"
      minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
```

[top](#)

### Complex Type: AlertCPoint

Super-types: None  
Sub-types:

- [AlertCMethod2Point](#) (by extension)
- [AlertCMethod4Point](#) (by extension)

**Name** AlertCPoint  
**Abstract** yes  
**Documentation** A single point on the road network defined by reference to a pre-defined ALERT-C location table and which has an associated direction of traffic flow.

#### XML Instance Representation

```
<...>
  <D2LogicalModel:alertCLocationCountryCode> D2LogicalModel:String </D2LogicalModel:alertCLocationCountryCode> [1] ?
  <D2LogicalModel:alertCLocationTableNumber> D2LogicalModel:String </D2LogicalModel:alertCLocationTableNumber> [1] ?
  <D2LogicalModel:alertCLocationTableVersion> D2LogicalModel:String </D2LogicalModel:alertCLocationTableVersion> [1] ?
  <D2LogicalModel:alertCPointExtension> D2LogicalModel:_ExtensionType </D2LogicalModel:alertCPointExtension> [0..1]
</...>
```

#### Schema Component Representation

```
<xs:complexType name="AlertCPoint" abstract="true">
  <xs:sequence>
    <xs:element name="alertCLocationCountryCode" type="D2LogicalModel:String" minOccurs="1" maxOccurs="1"/>
    <xs:element name="alertCLocationTableNumber" type="D2LogicalModel:String" minOccurs="1" maxOccurs="1"/>
    <xs:element name="alertCLocationTableVersion" type="D2LogicalModel:String" minOccurs="1" maxOccurs="1"/>
    <xs:element name="alertCPointExtension" type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
```

[top](#)

### Complex Type: AxleFlowValue

Super-types: [DataValue](#) < AxleFlowValue (by extension)

Sub-types: None

Name AxleFlowValue  
Abstract no  
Documentation A measured or calculated value of the flow rate of vehicle axles.

#### XML Instance Representation

```
<...  
  accuracy="D2LogicalModel:Percentage [0..1] ?"  
  computationalMethod="D2LogicalModel:ComputationMethodEnum [0..1] ?"  
  numberOfIncompleteInputs="D2LogicalModel:NonNegativeInteger [0..1] ?"  
  numberOfInputValuesUsed="D2LogicalModel:NonNegativeInteger [0..1] ?"  
  smoothingFactor="D2LogicalModel:Float [0..1] ?"  
  standardDeviation="D2LogicalModel:Float [0..1] ?"  
  supplierCalculatedDataQuality="D2LogicalModel:Percentage [0..1] ?">  
  <D2LogicalModel:dataError> D2LogicalModel:Boolean </D2LogicalModel:dataError> [0..1] ?  
  <D2LogicalModel:reasonForDataError> D2LogicalModel:MultilingualString </D2LogicalModel:reasonForDataError> [0..1]  
  ?  
  <D2LogicalModel:dataValueExtension> D2LogicalModel:_ExtensionType </D2LogicalModel:dataValueExtension> [0..1]  
  <D2LogicalModel:axleFlowRate> D2LogicalModel:AxlesPerHour </D2LogicalModel:axleFlowRate> [1] ?  
  <D2LogicalModel:axleFlowValueExtension> D2LogicalModel:_ExtensionType </D2LogicalModel:axleFlowValueExtension>  
  [0..1]  
</...>
```

#### Schema Component Representation

```
<xs:complexType name="AxleFlowValue">  
  <xs:complexContent>  
    <xs:extension base="D2LogicalModel:DataValue">  
      <xs:sequence>  
        <xs:element name="axleFlowRate" type="D2LogicalModel:AxlesPerHour" minOccurs="1" maxOccurs="1"/>  
        <xs:element name="axleFlowValueExtension" type="D2LogicalModel:_ExtensionType" minOccurs="0"/>  
      </xs:sequence>  
    </xs:extension>  
  </xs:complexContent>  
</xs:complexType>
```

[top](#)

### Complex Type: ConcentrationOfVehiclesValue

Super-types: [DataValue](#) < ConcentrationOfVehiclesValue (by extension)  
Sub-types: None

Name ConcentrationOfVehiclesValue  
Abstract no  
Documentation A measured or calculated value of the concentration of vehicles on a unit stretch of road in a given direction.

#### XML Instance Representation

```
<...  
  accuracy="D2LogicalModel:Percentage [0..1] ?"  
  computationalMethod="D2LogicalModel:ComputationMethodEnum [0..1] ?"  
  numberOfIncompleteInputs="D2LogicalModel:NonNegativeInteger [0..1] ?"  
  numberOfInputValuesUsed="D2LogicalModel:NonNegativeInteger [0..1] ?"  
  smoothingFactor="D2LogicalModel:Float [0..1] ?"  
  standardDeviation="D2LogicalModel:Float [0..1] ?"  
  supplierCalculatedDataQuality="D2LogicalModel:Percentage [0..1] ?">  
  <D2LogicalModel:dataError> D2LogicalModel:Boolean </D2LogicalModel:dataError> [0..1] ?  
  <D2LogicalModel:reasonForDataError> D2LogicalModel:MultilingualString </D2LogicalModel:reasonForDataError> [0..1]  
  ?  
  <D2LogicalModel:dataValueExtension> D2LogicalModel:_ExtensionType </D2LogicalModel:dataValueExtension> [0..1]  
  <D2LogicalModel:concentrationOfVehicles> D2LogicalModel:ConcentrationVehiclesPerKilometre  
  </D2LogicalModel:concentrationOfVehicles> [1] ?  
  <D2LogicalModel:concentrationOfVehiclesValueExtension> D2LogicalModel:_ExtensionType  
  </D2LogicalModel:concentrationOfVehiclesValueExtension> [0..1]  
</...>
```

#### Schema Component Representation

```
<xs:complexType name="ConcentrationOfVehiclesValue">  
  <xs:complexContent>  
    <xs:extension base="D2LogicalModel:DataValue">  
      <xs:sequence>  
        <xs:element name="concentrationOfVehicles" type="D2LogicalModel:ConcentrationVehiclesPerKilometre"  
          minOccurs="1" maxOccurs="1"/>  
        <xs:element name="concentrationOfVehiclesValueExtension" type="D2LogicalModel:_ExtensionType"  
          minOccurs="0"/>  
      </xs:sequence>  
    </xs:extension>  
  </xs:complexContent>  
</xs:complexType>
```

[top](#)

### Complex Type: D2LogicalModel

Super-types: None  
Sub-types: None

Name D2LogicalModel

**Abstract**

no

**Documentation**

The DATEX II logical model comprising exchange, content payload and management sub-models.

**XML Instance Representation**

```
<...
  modelBaseVersion="2 [1]">
  <D2LogicalModel:exchange> D2LogicalModel:Exchange </D2LogicalModel:exchange> [1]
  <D2LogicalModel:payloadPublication> D2LogicalModel:PayloadPublication </D2LogicalModel:payloadPublication> [0..1]
  <D2LogicalModel:d2LogicalModelExtension> D2LogicalModel:_ExtensionType </D2LogicalModel:d2LogicalModelExtension>
  [0..1]
</...>
```

**Schema Component Representation**

```
<xs:complexType name="D2LogicalModel">
  <xs:sequence>
    <xs:element name="exchange" type="D2LogicalModel:Exchange"/>
    <xs:element name="payloadPublication" type="D2LogicalModel:PayloadPublication" minOccurs="0"/>
    <xs:element name="d2LogicalModelExtension" type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
  </xs:sequence>
  <xs:attribute name="modelBaseVersion" use="required" fixed="2"/>
</xs:complexType>
```

[top](#)**Complex Type: DataValue****Super-types:** None**Sub-types:**

- [AxleFlowValue](#) (by extension)
- [ConcentrationOfVehiclesValue](#) (by extension)
- [DateTimeValue](#) (by extension)
- [DurationValue](#) (by extension)
- [OccupancyChangeValue](#) (by extension)
- [PcuFlowValue](#) (by extension)
- [TrafficStatusValue](#) (by extension)
- [VehicleCountValue](#) (by extension)
- [VehicleFlowValue](#) (by extension)

**Name** DataValue**Abstract** yes**Documentation**

A data value of something that can be measured or calculated. Any provided meta-data values specified in the attributes override any specified generic characteristics such as defined for a specific measurement in the MeasurementSiteTable.

**XML Instance Representation**

```
<...
  accuracy="D2LogicalModel:Percentage [0..1] ?"
  computationalMethod="D2LogicalModel:ComputationMethodEnum [0..1] ?"
  numberOfIncompleteInputs="D2LogicalModel:NonNegativeInteger [0..1] ?"
  numberOfInputValuesUsed="D2LogicalModel:NonNegativeInteger [0..1] ?"
  smoothingFactor="D2LogicalModel:Float [0..1] ?"
  standardDeviation="D2LogicalModel:Float [0..1] ?"
  supplierCalculatedDataQuality="D2LogicalModel:Percentage [0..1] ?">
  <D2LogicalModel:dataError> D2LogicalModel:Boolean </D2LogicalModel:dataError> [0..1] ?
  <D2LogicalModel:reasonForDataError> D2LogicalModel:MultilingualString </D2LogicalModel:reasonForDataError> [0..1]
  ?
  <D2LogicalModel:dataValueExtension> D2LogicalModel:_ExtensionType </D2LogicalModel:dataValueExtension> [0..1]
</...>
```

**Schema Component Representation**

```
<xs:complexType name="DataValue" abstract="true">
  <xs:sequence>
    <xs:element name="dataError" type="D2LogicalModel:Boolean" minOccurs="0" maxOccurs="1"/>
    <xs:element name="reasonForDataError" type="D2LogicalModel:MultilingualString" minOccurs="0" maxOccurs="1"/>
    <xs:element name="dataValueExtension" type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
  </xs:sequence>
  <xs:attribute name="accuracy" type="D2LogicalModel:Percentage" use="optional"/>
  <xs:attribute name="computationalMethod" type="D2LogicalModel:ComputationMethodEnum" use="optional"/>
  <xs:attribute name="numberOfIncompleteInputs" type="D2LogicalModel:NonNegativeInteger" use="optional"/>
  <xs:attribute name="numberOfInputValuesUsed" type="D2LogicalModel:NonNegativeInteger" use="optional"/>
  <xs:attribute name="smoothingFactor" type="D2LogicalModel:Float" use="optional"/>
  <xs:attribute name="standardDeviation" type="D2LogicalModel:Float" use="optional"/>
  <xs:attribute name="supplierCalculatedDataQuality" type="D2LogicalModel:Percentage" use="optional"/>
</xs:complexType>
```

[top](#)**Complex Type: DateTimeValue****Super-types:** [DataValue](#) < [DateTimeValue](#) (by extension)**Sub-types:** None**Name** DateTimeValue**Abstract** no**Documentation**

A measured or calculated value of an instance in time.

**XML Instance Representation**

```
<...
  accuracy="D2LogicalModel:Percentage [0..1] ?"
  computationalMethod="D2LogicalModel:ComputationMethodEnum [0..1] ?"
  </...>
```

```

numberOfIncompleteInputs="D2LogicalModel:NonNegativeInteger [0..1] ?"
numberOfInputValuesUsed="D2LogicalModel:NonNegativeInteger [0..1] ?"
smoothingFactor="D2LogicalModel:Float [0..1] ?"
standardDeviation="D2LogicalModel:Float [0..1] ?"
supplierCalculatedDataQuality="D2LogicalModel:Percentage [0..1] ?">
<D2LogicalModel:dataError> D2LogicalModel:Boolean </D2LogicalModel:dataError> [0..1] ?
<D2LogicalModel:reasonForDataError> D2LogicalModel:MultilingualString </D2LogicalModel:reasonForDataError> [0..1]
?
<D2LogicalModel:dataValueExtension> D2LogicalModel:ExtensionType </D2LogicalModel:dataValueExtension> [0..1]
<D2LogicalModel:dateTime> D2LogicalModel:DateTime </D2LogicalModel:dateTime> [1] ?
<D2LogicalModel:dateTimeValueExtension> D2LogicalModel:ExtensionType </D2LogicalModel:dateTimeValueExtension>
[0..1]
</...>

```

#### Schema Component Representation

```

<xs:complexType name="DateTimeValue">
  <xs:complexContent>
    <xs:extension base="D2LogicalModel:DataValue">
      <xs:sequence>
        <xs:element name="dateTime" type="D2LogicalModel:DateTime" minOccurs="1" maxOccurs="1"/>
        <xs:element name="dateTimeValueExtension" type="D2LogicalModel:ExtensionType" minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

```

[top](#)

### Complex Type: DistanceAlongLinearElement

Super-types: None

Sub-types:

- [DistanceFromLinearElementReferent](#) (by extension)
- [DistanceFromLinearElementStart](#) (by extension)
- [PercentageDistanceAlongLinearElement](#) (by extension)

**Name** DistanceAlongLinearElement  
**Abstract** yes  
**Documentation** Distance of a point along a linear element either measured from the start node or a defined referent on that linear element, where the start node is relative to the element definition rather than the direction of traffic flow.

#### XML Instance Representation

```

<...>
  <D2LogicalModel:distanceAlongLinearElementExtension> D2LogicalModel:ExtensionType
</D2LogicalModel:distanceAlongLinearElementExtension> [0..1]
</...>

```

#### Schema Component Representation

```

<xs:complexType name="DistanceAlongLinearElement" abstract="true">
  <xs:sequence>
    <xs:element name="distanceAlongLinearElementExtension" type="D2LogicalModel:ExtensionType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>

```

[top](#)

### Complex Type: DistanceFromLinearElementReferent

Super-types: [DistanceAlongLinearElement](#) < [DistanceFromLinearElementReferent](#) (by extension)

Sub-types: None

**Name** DistanceFromLinearElementReferent  
**Abstract** no  
**Documentation** Distance of a point along a linear element measured from a "from referent" on the linear element, in the sense relative to the linear element definition rather than the direction of traffic flow or optionally towards a "towards referent".

#### XML Instance Representation

```

<...>
  <D2LogicalModel:distanceAlongLinearElementExtension> D2LogicalModel:ExtensionType
</D2LogicalModel:distanceAlongLinearElementExtension> [0..1]
  <D2LogicalModel:distanceAlong> D2LogicalModel:MetresAsFloat </D2LogicalModel:distanceAlong> [1] ?
  <D2LogicalModel:fromReferent> D2LogicalModel:Referent </D2LogicalModel:fromReferent> [1] ?
  <D2LogicalModel:towardsReferent> D2LogicalModel:Referent </D2LogicalModel:towardsReferent> [0..1] ?
  <D2LogicalModel:distanceFromLinearElementReferentExtension> D2LogicalModel:ExtensionType
</D2LogicalModel:distanceFromLinearElementReferentExtension> [0..1]
</...>

```

#### Schema Component Representation

```

<xs:complexType name="DistanceFromLinearElementReferent">
  <xs:complexContent>
    <xs:extension base="D2LogicalModel:DistanceAlongLinearElement">
      <xs:sequence>
        <xs:element name="distanceAlong" type="D2LogicalModel:MetresAsFloat" minOccurs="1" maxOccurs="1"/>
        <xs:element name="fromReferent" type="D2LogicalModel:Referent"/>
        <xs:element name="towardsReferent" type="D2LogicalModel:Referent" minOccurs="0"/>
        <xs:element name="distanceFromLinearElementReferentExtension" type="D2LogicalModel:ExtensionType"
minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

```

```
</xs:extension>
</xs:complexContent>
</xs:complexType>
```

[top](#)

## Complex Type: **DistanceFromLinearElementStart**

Super-types: [DistanceAlongLinearElement](#) < **DistanceFromLinearElementStart** (by extension)  
Sub-types: None

**Name** DistanceFromLinearElementStart  
**Abstract** no  
**Documentation** Distance of a point along a linear element measured from the start node of the linear element, where start node is relative to the element definition rather than the direction of traffic flow.

### XML Instance Representation

```
<...>
  <D2LogicalModel:distanceAlongLinearElementExtension> D2LogicalModel:_ExtensionType
</D2LogicalModel:distanceAlongLinearElementExtension> [0..1]
  <D2LogicalModel:distanceAlong> D2LogicalModel:MetresAsFloat </D2LogicalModel:distanceAlong> [1] ?
  <D2LogicalModel:distanceFromLinearElementStartExtension> D2LogicalModel:_ExtensionType
</D2LogicalModel:distanceFromLinearElementStartExtension> [0..1]
</...>
```

### Schema Component Representation

```
<xs:complexType name="DistanceFromLinearElementStart">
  <xs:complexContent>
    <xs:extension base="D2LogicalModel:DistanceAlongLinearElement">
      <xs:sequence>
        <xs:element name="distanceAlong" type="D2LogicalModel:MetresAsFloat" minOccurs="1" maxOccurs="1"/>
        <xs:element name="distanceFromLinearElementStartExtension" type="D2LogicalModel:_ExtensionType"
          minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

[top](#)

## Complex Type: **DurationValue**

Super-types: [DataValue](#) < **DurationValue** (by extension)  
Sub-types: None

**Name** DurationValue  
**Abstract** no  
**Documentation** A measured or calculated value of a period of time.

### XML Instance Representation

```
<...
accuracy="D2LogicalModel:Percentage [0..1] ?"
computationalMethod="D2LogicalModel:ComputationMethodEnum [0..1] ?"
numberOfIncompleteInputs="D2LogicalModel:NonNegativeInteger [0..1] ?"
numberOfInputValuesUsed="D2LogicalModel:NonNegativeInteger [0..1] ?"
smoothingFactor="D2LogicalModel:Float [0..1] ?"
standardDeviation="D2LogicalModel:Float [0..1] ?"
supplierCalculatedDataQuality="D2LogicalModel:Percentage [0..1] ?">
  <D2LogicalModel:dataError> D2LogicalModel:Boolean </D2LogicalModel:dataError> [0..1] ?
  <D2LogicalModel:reasonForDataError> D2LogicalModel:MultilingualString </D2LogicalModel:reasonForDataError> [0..1]
  ?
  <D2LogicalModel:dataValueExtension> D2LogicalModel:_ExtensionType </D2LogicalModel:dataValueExtension> [0..1]
  <D2LogicalModel:duration> D2LogicalModel:Seconds </D2LogicalModel:duration> [1] ?
  <D2LogicalModel:durationValueExtension> D2LogicalModel:_ExtensionType </D2LogicalModel:durationValueExtension>
  [0..1]
</...>
```

### Schema Component Representation

```
<xs:complexType name="DurationValue">
  <xs:complexContent>
    <xs:extension base="D2LogicalModel:DataValue">
      <xs:sequence>
        <xs:element name="duration" type="D2LogicalModel:Seconds" minOccurs="1" maxOccurs="1"/>
        <xs:element name="durationValueExtension" type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

[top](#)

## Complex Type: **ElaboratedDataFault**

Super-types: [Fault](#) < **ElaboratedDataFault** (by extension)  
Sub-types: None

**Name** ElaboratedDataFault

**Abstract** no  
**Documentation** Details of a fault which is being reported for the related elaborated data.

**XML Instance Representation**

```
<...>
  <D2LogicalModel:faultIdentifier> D2LogicalModel:String </D2LogicalModel:faultIdentifier> [0..1] ?
  <D2LogicalModel:faultDescription> D2LogicalModel:String </D2LogicalModel:faultDescription> [0..1] ?
  <D2LogicalModel:faultCreationTime> D2LogicalModel:DateTime </D2LogicalModel:faultCreationTime> [0..1] ?
  <D2LogicalModel:faultLastUpdateTime> D2LogicalModel:DateTime </D2LogicalModel:faultLastUpdateTime> [1] ?
  <D2LogicalModel:faultSeverity> D2LogicalModel:FaultSeverityEnum </D2LogicalModel:faultSeverity> [0..1] ?
  <D2LogicalModel:faultExtension> D2LogicalModel:_ExtensionType </D2LogicalModel:faultExtension> [0..1]
  <D2LogicalModel:elaboratedDataFault> D2LogicalModel:ElaboratedDataFaultEnum </D2LogicalModel:elaboratedDataFault>
  [1] ?
  <D2LogicalModel:elaboratedDataFaultExtension> D2LogicalModel:_ExtensionType
</D2LogicalModel:elaboratedDataFaultExtension> [0..1]
</...>
```

**Schema Component Representation**

```
<xs:complexType name="ElaboratedDataFault">
  <xs:complexContent>
    <xs:extension base="D2LogicalModel:Fault">
      <xs:sequence>
        <xs:element name="elaboratedDataFault" type="D2LogicalModel:ElaboratedDataFaultEnum" minOccurs="1"
          maxOccurs="1"/>
        <xs:element name="elaboratedDataFaultExtension" type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

[top](#)

**Complex Type: Exchange**

Super-types:	None
Sub-types:	None

**Name** Exchange  
**Abstract** no  
**Documentation** Details associated with the management of the exchange between the supplier and the client.

**XML Instance Representation**

```
<...>
  <D2LogicalModel:supplierIdentification> D2LogicalModel:InternationalIdentifier
</D2LogicalModel:supplierIdentification> [1]
  <D2LogicalModel:exchangeExtension> D2LogicalModel:_ExtensionType </D2LogicalModel:exchangeExtension> [0..1]
</...>
```

**Schema Component Representation**

```
<xs:complexType name="Exchange">
  <xs:sequence>
    <xs:element name="supplierIdentification" type="D2LogicalModel:InternationalIdentifier"/>
    <xs:element name="exchangeExtension" type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
```

[top](#)

**Complex Type: Fault**

Super-types:	None
Sub-types:	<ul style="list-style-type: none"> <li>• <a href="#">ElaboratedDataFault</a> (by extension)</li> <li>• <a href="#">MeasurementEquipmentFault</a> (by extension)</li> <li>• <a href="#">VmsFault</a> (by extension)</li> <li>• <a href="#">VmsUnitFault</a> (by extension)</li> </ul>

**Name** Fault  
**Abstract** no  
**Documentation** Information about a fault relating to a specific piece of equipment or process.

**XML Instance Representation**

```
<...>
  <D2LogicalModel:faultIdentifier> D2LogicalModel:String </D2LogicalModel:faultIdentifier> [0..1] ?
  <D2LogicalModel:faultDescription> D2LogicalModel:String </D2LogicalModel:faultDescription> [0..1] ?
  <D2LogicalModel:faultCreationTime> D2LogicalModel:DateTime </D2LogicalModel:faultCreationTime> [0..1] ?
  <D2LogicalModel:faultLastUpdateTime> D2LogicalModel:DateTime </D2LogicalModel:faultLastUpdateTime> [1] ?
  <D2LogicalModel:faultSeverity> D2LogicalModel:FaultSeverityEnum </D2LogicalModel:faultSeverity> [0..1] ?
  <D2LogicalModel:faultExtension> D2LogicalModel:_ExtensionType </D2LogicalModel:faultExtension> [0..1]
</...>
```

**Schema Component Representation**

```
<xs:complexType name="Fault">
  <xs:sequence>
    <xs:element name="faultIdentifier" type="D2LogicalModel:String" minOccurs="0" maxOccurs="1"/>
    <xs:element name="faultDescription" type="D2LogicalModel:String" minOccurs="0" maxOccurs="1"/>
    <xs:element name="faultCreationTime" type="D2LogicalModel:DateTime" minOccurs="0" maxOccurs="1"/>
    <xs:element name="faultLastUpdateTime" type="D2LogicalModel:DateTime" minOccurs="1" maxOccurs="1"/>
  </xs:sequence>
</xs:complexType>
```

```

<xs:element name="faultSeverity" type="D2LogicalModel:FaultSeverityEnum" minOccurs="0" maxOccurs="1"/>
<xs:element name="faultExtension" type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
</xs:sequence>
</xs:complexType>

```

[top](#)

## Complex Type: GroupOfLocations

Super-types:	None
Sub-types:	<ul style="list-style-type: none"> <li>• <a href="#">Location</a> (by extension) <ul style="list-style-type: none"> <li>◦ <a href="#">NetworkLocation</a> (by extension) <ul style="list-style-type: none"> <li>▪ <a href="#">Linear</a> (by extension)</li> <li>▪ <a href="#">Point</a> (by extension)</li> </ul> </li> </ul> </li> </ul>

<b>Name</b>	GroupOfLocations
<b>Abstract</b>	yes
<b>Documentation</b>	One or more physically separate locations. Multiple locations may be related, as in an itinerary (or route), or may be unrelated. It is not for identifying the same physical location using different Location objects for different referencing systems.

### XML Instance Representation

```

<...>
<D2LogicalModel:groupOfLocationsExtension> D2LogicalModel:_ExtensionType
</D2LogicalModel:groupOfLocationsExtension> [0..1]
</...>

```

### Schema Component Representation

```

<xs:complexType name="GroupOfLocations" abstract="true">
<xs:sequence>
<xs:element name="groupOfLocationsExtension" type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
</xs:sequence>
</xs:complexType>

```

[top](#)

## Complex Type: HeaderInformation

Super-types:	None
Sub-types:	None

<b>Name</b>	HeaderInformation
<b>Abstract</b>	no
<b>Documentation</b>	Management information relating to the data contained within a publication.

### XML Instance Representation

```

<...>
<D2LogicalModel:areaOfInterest> D2LogicalModel:AreaOfInterestEnum </D2LogicalModel:areaOfInterest> [0..1] ?
<D2LogicalModel:confidentiality> D2LogicalModel:ConfidentialityValueEnum </D2LogicalModel:confidentiality> [1] ?
<D2LogicalModel:informationStatus> D2LogicalModel:InformationStatusEnum </D2LogicalModel:informationStatus> [1] ?
<D2LogicalModel:urgency> D2LogicalModel:UrgencyEnum </D2LogicalModel:urgency> [0..1] ?
<D2LogicalModel:headerInformationExtension> D2LogicalModel:_ExtensionType
</D2LogicalModel:headerInformationExtension> [0..1]
</...>

```

### Schema Component Representation

```

<xs:complexType name="HeaderInformation">
<xs:sequence>
<xs:element name="areaOfInterest" type="D2LogicalModel:AreaOfInterestEnum" minOccurs="0" maxOccurs="1"/>
<xs:element name="confidentiality" type="D2LogicalModel:ConfidentialityValueEnum" minOccurs="1" maxOccurs="1"/>
<xs:element name="informationStatus" type="D2LogicalModel:InformationStatusEnum" minOccurs="1" maxOccurs="1"/>
<xs:element name="urgency" type="D2LogicalModel:UrgencyEnum" minOccurs="0" maxOccurs="1"/>
<xs:element name="headerInformationExtension" type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
</xs:sequence>
</xs:complexType>

```

[top](#)

## Complex Type: InternationalIdentifier

Super-types:	None
Sub-types:	None

<b>Name</b>	InternationalIdentifier
<b>Abstract</b>	no
<b>Documentation</b>	An identifier/name whose range is specific to the particular country.

### XML Instance Representation

```

<...>
<D2LogicalModel:country> D2LogicalModel:CountryEnum </D2LogicalModel:country> [1] ?
<D2LogicalModel:nationalIdentifier> D2LogicalModel:String </D2LogicalModel:nationalIdentifier> [1] ?
<D2LogicalModel:internationalIdentifierExtension> D2LogicalModel:_ExtensionType
</D2LogicalModel:internationalIdentifierExtension> [0..1]
</...>

```



## Schema Component Representation

```
<xs:complexType name="InternationalIdentifier">
  <xs:sequence>
    <xs:element name="country" type="D2LogicalModel:CountryEnum" minOccurs="1" maxOccurs="1"/>
    <xs:element name="nationalIdentifier" type="D2LogicalModel:String" minOccurs="1" maxOccurs="1"/>
    <xs:element name="internationalIdentifierExtension" type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
```

[top](#)

## Complex Type: Junction

Super-types: None

Sub-types: None

Name Junction

Abstract no

Documentation Junction (on a highway), can also be an interchange or if applicable also a motorway service station (see junctionClassification).

## XML Instance Representation

```
<...>
  <D2LogicalModel:junctionClassification> D2LogicalModel:JunctionClassificationEnum
</D2LogicalModel:junctionClassification> [0..1] ?
  <D2LogicalModel:junctionName> D2LogicalModel:MultilingualString </D2LogicalModel:junctionName> [1] ?
  <D2LogicalModel:junctionNumber> D2LogicalModel:String </D2LogicalModel:junctionNumber> [0..1] ?
  <D2LogicalModel:motorway> D2LogicalModel:Road </D2LogicalModel:motorway> [0..1] ?
  <D2LogicalModel:destinationMotorway> D2LogicalModel:Road </D2LogicalModel:destinationMotorway> [0..*] ?
  <D2LogicalModel:junctionExtension> D2LogicalModel:_ExtensionType </D2LogicalModel:junctionExtension> [0..1]
</...>
```

## Schema Component Representation

```
<xs:complexType name="Junction">
  <xs:sequence>
    <xs:element name="junctionClassification" type="D2LogicalModel:JunctionClassificationEnum" minOccurs="0" maxOccurs="1"/>
    <xs:element name="junctionName" type="D2LogicalModel:MultilingualString" minOccurs="1" maxOccurs="1"/>
    <xs:element name="junctionNumber" type="D2LogicalModel:String" minOccurs="0" maxOccurs="1"/>
    <xs:element name="motorway" type="D2LogicalModel:Road" minOccurs="0"/>
    <xs:element name="destinationMotorway" type="D2LogicalModel:Road" minOccurs="0" maxOccurs="unbounded"/>
    <xs:element name="junctionExtension" type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
```

[top](#)

## Complex Type: Linear

Super-types: [GroupOfLocations](#) < [Location](#) (by extension) < [NetworkLocation](#) (by extension) < **Linear** (by extension)

Sub-types: None

Name Linear

Abstract no

Documentation A linear section along a single road with optional directionality defined between two points on the same road.

## XML Instance Representation

```
<...>
  <D2LogicalModel:groupOfLocationsExtension> D2LogicalModel:_ExtensionType
</D2LogicalModel:groupOfLocationsExtension> [0..1]
  <D2LogicalModel:locationForDisplay> D2LogicalModel:PointCoordinates </D2LogicalModel:locationForDisplay> [0..1] ?
  <D2LogicalModel:locationExtension> D2LogicalModel:_ExtensionType </D2LogicalModel:locationExtension> [0..1]
  <D2LogicalModel:supplementaryPositionalDescription> D2LogicalModel:SupplementaryPositionalDescription
</D2LogicalModel:supplementaryPositionalDescription> [0..1]
  <D2LogicalModel:networkLocationExtension> D2LogicalModel:_ExtensionType </D2LogicalModel:networkLocationExtension> [0..1]
  <D2LogicalModel:tpegLinearLocation> D2LogicalModel:TpegLinearLocation </D2LogicalModel:tpegLinearLocation> [0..1]
  <D2LogicalModel:alertCLinear> D2LogicalModel:AlertCLinear </D2LogicalModel:alertCLinear> [0..1]
  <D2LogicalModel:linearWithinLinearElement> D2LogicalModel:LinearWithinLinearElement
</D2LogicalModel:linearWithinLinearElement> [0..1]
  <D2LogicalModel:linearExtension> D2LogicalModel:_LinearExtensionType </D2LogicalModel:linearExtension> [0..1]
</...>
```

## Schema Component Representation

```
<xs:complexType name="Linear">
  <xs:complexContent>
    <xs:extension base="D2LogicalModel:NetworkLocation">
      <xs:sequence>
        <xs:element name="tpegLinearLocation" type="D2LogicalModel:TpegLinearLocation" minOccurs="0"/>
        <xs:element name="alertCLinear" type="D2LogicalModel:AlertCLinear" minOccurs="0"/>
        <xs:element name="linearWithinLinearElement" type="D2LogicalModel:LinearWithinLinearElement" minOccurs="0"/>
        <xs:element name="linearExtension" type="D2LogicalModel:_LinearExtensionType" minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

[top](#)



## Complex Type: LinearElement

Super-types:	None
Sub-types:	<ul style="list-style-type: none"><li>• <a href="#">LinearElementByCode</a> (by extension)</li><li>• <a href="#">LinearElementByPoints</a> (by extension)</li></ul>

Name	LinearElement
Abstract	no
Documentation	A linear element along a single linear object, consistent with ISO 19148 definitions.

### XML Instance Representation

```
<...>
  <D2LogicalModel:roadName> D2LogicalModel:MultilingualString </D2LogicalModel:roadName> [0..1] ?
  <D2LogicalModel:roadNumber> D2LogicalModel:String </D2LogicalModel:roadNumber> [0..1] ?
  <D2LogicalModel:linearElementReferenceModel> D2LogicalModel:String </D2LogicalModel:linearElementReferenceModel>
  [0..1] ?
  <D2LogicalModel:linearElementReferenceModelVersion> D2LogicalModel:String
  </D2LogicalModel:linearElementReferenceModelVersion> [0..1] ?
  <D2LogicalModel:linearElementNature> D2LogicalModel:LinearElementNatureEnum </D2LogicalModel:linearElementNature>
  [0..1] ?
  <D2LogicalModel:linearElementExtension> D2LogicalModel:ExtensionType </D2LogicalModel:linearElementExtension>
  [0..1]
</...>
```

### Schema Component Representation

```
<xs:complexType name="LinearElement">
  <xs:sequence>
    <xs:element name="roadName" type="D2LogicalModel:MultilingualString" minOccurs="0" maxOccurs="1"/>
    <xs:element name="roadNumber" type="D2LogicalModel:String" minOccurs="0" maxOccurs="1"/>
    <xs:element name="linearElementReferenceModel" type="D2LogicalModel:String" minOccurs="0" maxOccurs="1"/>
    <xs:element name="linearElementReferenceModelVersion" type="D2LogicalModel:String" minOccurs="0" maxOccurs="1"/>
    <xs:element name="linearElementNature" type="D2LogicalModel:LinearElementNatureEnum" minOccurs="0"
    maxOccurs="1"/>
    <xs:element name="linearElementExtension" type="D2LogicalModel:ExtensionType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
```

[top](#)

## Complex Type: LinearElementByCode

Super-types:	<a href="#">LinearElement</a> < <a href="#">LinearElementByCode</a> (by extension)
Sub-types:	None

Name	LinearElementByCode
Abstract	no
Documentation	A linear element along a single linear object defined by its identifier or code in a road network reference model (specified in LinearElement class) which segments the road network according to specific business rules.

### XML Instance Representation

```
<...>
  <D2LogicalModel:roadName> D2LogicalModel:MultilingualString </D2LogicalModel:roadName> [0..1] ?
  <D2LogicalModel:roadNumber> D2LogicalModel:String </D2LogicalModel:roadNumber> [0..1] ?
  <D2LogicalModel:linearElementReferenceModel> D2LogicalModel:String </D2LogicalModel:linearElementReferenceModel>
  [0..1] ?
  <D2LogicalModel:linearElementReferenceModelVersion> D2LogicalModel:String
  </D2LogicalModel:linearElementReferenceModelVersion> [0..1] ?
  <D2LogicalModel:linearElementNature> D2LogicalModel:LinearElementNatureEnum </D2LogicalModel:linearElementNature>
  [0..1] ?
  <D2LogicalModel:linearElementExtension> D2LogicalModel:ExtensionType </D2LogicalModel:linearElementExtension>
  [0..1]
  <D2LogicalModel:linearElementIdentifier> D2LogicalModel:String </D2LogicalModel:linearElementIdentifier> [1] ?
  <D2LogicalModel:linearElementByCodeExtension> D2LogicalModel:ExtensionType
  </D2LogicalModel:linearElementByCodeExtension> [0..1]
</...>
```

### Schema Component Representation

```
<xs:complexType name="LinearElementByCode">
  <xs:complexContent>
    <xs:extension base="D2LogicalModel:LinearElement">
      <xs:sequence>
        <xs:element name="linearElementIdentifier" type="D2LogicalModel:String" minOccurs="1" maxOccurs="1"/>
        <xs:element name="linearElementByCodeExtension" type="D2LogicalModel:ExtensionType" minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

[top](#)

## Complex Type: LinearElementByPoints

Super-types:	<a href="#">LinearElement</a> < <a href="#">LinearElementByPoints</a> (by extension)
Sub-types:	None

Name	LinearElementByPoints
------	-----------------------

**Abstract** no  
**Documentation** A linear element along a single linear object defined by its start and end points.

**XML Instance Representation**

```
<...>
  <D2LogicalModel:roadName> D2LogicalModel:MultilingualString </D2LogicalModel:roadName> [0..1] ?
  <D2LogicalModel:roadNumber> D2LogicalModel:String </D2LogicalModel:roadNumber> [0..1] ?
  <D2LogicalModel:linearElementReferenceModel> D2LogicalModel:String </D2LogicalModel:linearElementReferenceModel>
  [0..1] ?
  <D2LogicalModel:linearElementReferenceModelVersion> D2LogicalModel:String
  </D2LogicalModel:linearElementReferenceModelVersion> [0..1] ?
  <D2LogicalModel:linearElementNature> D2LogicalModel:LinearElementNatureEnum </D2LogicalModel:linearElementNature>
  [0..1] ?
  <D2LogicalModel:linearElementExtension> D2LogicalModel:_ExtensionType </D2LogicalModel:linearElementExtension>
  [0..1]
  <D2LogicalModel:startPointOfLinearElement> D2LogicalModel:Referent </D2LogicalModel:startPointOfLinearElement> [1]
  ?
  <D2LogicalModel:intermediatePointOnLinearElement> D2LogicalModel:_IntermediatePointOnLinearElement
  </D2LogicalModel:intermediatePointOnLinearElement> [0..*] ?
  <D2LogicalModel:endPointOfLinearElement> D2LogicalModel:Referent </D2LogicalModel:endPointOfLinearElement> [1] ?
  <D2LogicalModel:linearElementByPointsExtension> D2LogicalModel:_ExtensionType
  </D2LogicalModel:linearElementByPointsExtension> [0..1]
</...>
```

**Schema Component Representation**

```
<xs:complexType name="LinearElementByPoints">
  <xs:complexContent>
    <xs:extension base="D2LogicalModel:LinearElement">
      <xs:sequence>
        <xs:element name="startPointOfLinearElement" type="D2LogicalModel:Referent"/>
        <xs:element name="intermediatePointOnLinearElement" type="D2LogicalModel:_IntermediatePointOnLinearElement"
        minOccurs="0" maxOccurs="unbounded"/>
        <xs:element name="endPointOfLinearElement" type="D2LogicalModel:Referent"/>
        <xs:element name="linearElementByPointsExtension" type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

[top](#)

**Complex Type: LinearWithinLinearElement**

Super-types:	None
Sub-types:	None

**Name** LinearWithinLinearElement  
**Abstract** no  
**Documentation** A linear section along a linear element where the linear element is either a part of or the whole of a linear object (i.e. a road), consistent with ISO 19148 definitions.

**XML Instance Representation**

```
<...>
  <D2LogicalModel:administrativeAreaOfLinearSection> D2LogicalModel:MultilingualString
  </D2LogicalModel:administrativeAreaOfLinearSection> [0..1] ?
  <D2LogicalModel:directionBoundOnLinearSection> D2LogicalModel:DirectionEnum
  </D2LogicalModel:directionBoundOnLinearSection> [0..1] ?
  <D2LogicalModel:directionRelativeOnLinearSection> D2LogicalModel:LinearReferencingDirectionEnum
  </D2LogicalModel:directionRelativeOnLinearSection> [0..1] ?
  <D2LogicalModel:heightGradeOfLinearSection> D2LogicalModel:HeightGradeEnum
  </D2LogicalModel:heightGradeOfLinearSection> [0..1] ?
  <D2LogicalModel:linearElement> D2LogicalModel:LinearElement </D2LogicalModel:linearElement> [1]
  <D2LogicalModel:fromPoint> D2LogicalModel:DistanceAlongLinearElement </D2LogicalModel:fromPoint> [1] ?
  <D2LogicalModel:toPoint> D2LogicalModel:DistanceAlongLinearElement </D2LogicalModel:toPoint> [1] ?
  <D2LogicalModel:linearWithinLinearElementExtension> D2LogicalModel:_ExtensionType
  </D2LogicalModel:linearWithinLinearElementExtension> [0..1]
</...>
```

**Schema Component Representation**

```
<xs:complexType name="LinearWithinLinearElement">
  <xs:sequence>
    <xs:element name="administrativeAreaOfLinearSection" type="D2LogicalModel:MultilingualString" minOccurs="0"
    maxOccurs="1"/>
    <xs:element name="directionBoundOnLinearSection" type="D2LogicalModel:DirectionEnum" minOccurs="0"
    maxOccurs="1"/>
    <xs:element name="directionRelativeOnLinearSection" type="D2LogicalModel:LinearReferencingDirectionEnum"
    minOccurs="0" maxOccurs="1"/>
    <xs:element name="heightGradeOfLinearSection" type="D2LogicalModel:HeightGradeEnum" minOccurs="0"
    maxOccurs="1"/>
    <xs:element name="linearElement" type="D2LogicalModel:LinearElement"/>
    <xs:element name="fromPoint" type="D2LogicalModel:DistanceAlongLinearElement"/>
    <xs:element name="toPoint" type="D2LogicalModel:DistanceAlongLinearElement"/>
    <xs:element name="linearWithinLinearElementExtension" type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
```

[top](#)

**Complex Type: Location**

Super-types:	<a href="#">GroupOfLocations</a> < Location (by extension)
Sub-types:	

- [NetworkLocation](#) (by extension)
  - [Linear](#) (by extension)
  - [Point](#) (by extension)

<b>Name</b>	Location
<b>Abstract</b>	yes
<b>Documentation</b>	The specification of a location either on a network (as a point or a linear location) or as an area. This may be provided in one or more referencing systems.

#### XML Instance Representation

```
<...>
  <D2LogicalModel:groupOfLocationsExtension> D2LogicalModel: ExtensionType
</D2LogicalModel:groupOfLocationsExtension> [0..1]
  <D2LogicalModel:locationForDisplay> D2LogicalModel:PointCoordinates </D2LogicalModel:locationForDisplay> [0..1] ?
  <D2LogicalModel:locationExtension> D2LogicalModel: ExtensionType </D2LogicalModel:locationExtension> [0..1]
</...>
```

#### Schema Component Representation

```
<xs:complexType name="Location" abstract="true">
  <xs:complexContent>
    <xs:extension base="D2LogicalModel:GroupOfLocations">
      <xs:sequence>
        <xs:element name="locationForDisplay" type="D2LogicalModel:PointCoordinates" minOccurs="0"/>
        <xs:element name="locationExtension" type="D2LogicalModel: ExtensionType" minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

[top](#)

### Complex Type: **MeasurementEquipmentFault**

<b>Super-types:</b>	<a href="#">Fault</a> < <a href="#">MeasurementEquipmentFault</a> (by extension)
<b>Sub-types:</b>	None

<b>Name</b>	MeasurementEquipmentFault
<b>Abstract</b>	no
<b>Documentation</b>	Details of a fault which is being reported for the related measurement equipment.

#### XML Instance Representation

```
<...>
  <D2LogicalModel:faultIdentifier> D2LogicalModel:String </D2LogicalModel:faultIdentifier> [0..1] ?
  <D2LogicalModel:faultDescription> D2LogicalModel:String </D2LogicalModel:faultDescription> [0..1] ?
  <D2LogicalModel:faultCreationTime> D2LogicalModel:DateTime </D2LogicalModel:faultCreationTime> [0..1] ?
  <D2LogicalModel:faultLastUpdateTime> D2LogicalModel:DateTime </D2LogicalModel:faultLastUpdateTime> [1] ?
  <D2LogicalModel:faultSeverity> D2LogicalModel:FaultSeverityEnum </D2LogicalModel:faultSeverity> [0..1] ?
  <D2LogicalModel:faultExtension> D2LogicalModel: ExtensionType </D2LogicalModel:faultExtension> [0..1]
  <D2LogicalModel:measurementEquipmentFault> D2LogicalModel:MeasurementEquipmentFaultEnum
  </D2LogicalModel:measurementEquipmentFault> [1] ?
  <D2LogicalModel:measurementEquipmentFaultExtension> D2LogicalModel: ExtensionType
  </D2LogicalModel:measurementEquipmentFaultExtension> [0..1]
</...>
```

#### Schema Component Representation

```
<xs:complexType name="MeasurementEquipmentFault">
  <xs:complexContent>
    <xs:extension base="D2LogicalModel:Fault">
      <xs:sequence>
        <xs:element name="measurementEquipmentFault" type="D2LogicalModel:MeasurementEquipmentFaultEnum"
          minOccurs="1" maxOccurs="1"/>
        <xs:element name="measurementEquipmentFaultExtension" type="D2LogicalModel: ExtensionType" minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

[top](#)

### Complex Type: **MultilingualString**

<b>Super-types:</b>	None
<b>Sub-types:</b>	None

<b>Name</b>	MultilingualString
<b>Abstract</b>	no

#### XML Instance Representation

```
<...>
  <D2LogicalModel:values> [1]
    <D2LogicalModel:value> D2LogicalModel:MultilingualStringValue </D2LogicalModel:value> [1..*]
  </D2LogicalModel:values>
</...>
```

#### Schema Component Representation

```
<xs:complexType name="MultilingualString">
  <xs:sequence>
    <xs:element name="values">
```

```

    <xs:complexType>
      <xs:sequence>
        <xs:element name="value" type="D2LogicalModel:MultilingualStringValue" maxOccurs="unbounded" />
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:sequence>
</xs:complexType>

```

[top](#)

## Complex Type: MultilingualStringValue

Super-types: [xs:string](#) < [MultilingualStringValueType](#) (by restriction) < **MultilingualStringValue** (by extension)

Sub-types: None

Name MultilingualStringValue

**Abstract** no

### XML Instance Representation

```

<...
  lang="xs:language [0..1]">
    D2LogicalModel:MultilingualStringValueType
  </...>

```

### Schema Component Representation

```

<xs:complexType name="MultilingualStringValue">
  <xs:simpleContent>
    <xs:extension base="D2LogicalModel:MultilingualStringValueType">
      <xs:attribute name="lang" type="xs:language" />
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>

```

[top](#)

## Complex Type: NetworkLocation

Super-types: [GroupOfLocations](#) < [Location](#) (by extension) < **NetworkLocation** (by extension)

Sub-types:

- [Linear](#) (by extension)
- [Point](#) (by extension)

Name NetworkLocation

**Abstract** yes

Documentation The specification of a location on a network (as a point or a linear location).

### XML Instance Representation

```

<...>
  <D2LogicalModel:groupOfLocationsExtension> D2LogicalModel:_ExtensionType
</D2LogicalModel:groupOfLocationsExtension> [0..1]
  <D2LogicalModel:locationForDisplay> D2LogicalModel:PointCoordinates </D2LogicalModel:locationForDisplay> [0..1] ?
  <D2LogicalModel:locationExtension> D2LogicalModel:_ExtensionType </D2LogicalModel:locationExtension> [0..1]
  <D2LogicalModel:supplementaryPositionalDescription> D2LogicalModel:SupplementaryPositionalDescription
</D2LogicalModel:supplementaryPositionalDescription> [0..1]
  <D2LogicalModel:networkLocationExtension> D2LogicalModel:_ExtensionType </D2LogicalModel:networkLocationExtension>
  [0..1]
</...>

```

### Schema Component Representation

```

<xs:complexType name="NetworkLocation" abstract="true">
  <xs:complexContent>
    <xs:extension base="D2LogicalModel:Location">
      <xs:sequence>
        <xs:element name="supplementaryPositionalDescription"
          type="D2LogicalModel:SupplementaryPositionalDescription" minOccurs="0"/>
        <xs:element name="networkLocationExtension" type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

```

[top](#)

## Complex Type: OccupancyChangeValue

Super-types: [DataValue](#) < **OccupancyChangeValue** (by extension)

Sub-types: None

Name OccupancyChangeValue

**Abstract** no

Documentation A measured or calculated value of change of occupied parking spaces expressed as integer.

### XML Instance Representation

```

<...
  accuracy="D2LogicalModel:Percentage [0..1] ?"
  computationalMethod="D2LogicalModel:ComputationMethodEnum [0..1] ?"

```

```

numberOfIncompleteInputs="D2LogicalModel:NonNegativeInteger [0..1] ?"
numberOfInputValuesUsed="D2LogicalModel:NonNegativeInteger [0..1] ?"
smoothingFactor="D2LogicalModel:Float [0..1] ?"
standardDeviation="D2LogicalModel:Float [0..1] ?"
supplierCalculatedDataQuality="D2LogicalModel:Percentage [0..1] ?">
<D2LogicalModel:dataError> D2LogicalModel:Boolean </D2LogicalModel:dataError> [0..1] ?
<D2LogicalModel:reasonForDataError> D2LogicalModel:MultilingualString </D2LogicalModel:reasonForDataError> [0..1]
?
<D2LogicalModel:dataValueExtension> D2LogicalModel:_ExtensionType </D2LogicalModel:dataValueExtension> [0..1]
<D2LogicalModel:occupancyChange> D2LogicalModel:Integer </D2LogicalModel:occupancyChange> [1] ?
<D2LogicalModel:occupancyChangeValueExtension> D2LogicalModel:_ExtensionType
</D2LogicalModel:occupancyChangeValueExtension> [0..1]
</...>

```

#### Schema Component Representation

```

<xs:complexType name="OccupancyChangeValue">
  <xs:complexContent>
    <xs:extension base="D2LogicalModel:DataValue">
      <xs:sequence>
        <xs:element name="occupancyChange" type="D2LogicalModel:Integer" minOccurs="1" maxOccurs="1"/>
        <xs:element name="occupancyChangeValueExtension" type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

```

[top](#)

### Complex Type: OffsetDistance

Super-types: None  
Sub-types: None

**Name** OffsetDistance  
**Abstract** no  
**Documentation** The non negative offset distance from the ALERT-C referenced point to the actual point.

#### XML Instance Representation

```

<...>
<D2LogicalModel:offsetDistance> D2LogicalModel:MetresAsNonNegativeInteger </D2LogicalModel:offsetDistance> [1] ?
<D2LogicalModel:offsetDistanceExtension> D2LogicalModel:_ExtensionType </D2LogicalModel:offsetDistanceExtension>
[0..1]
</...>

```

#### Schema Component Representation

```

<xs:complexType name="OffsetDistance">
  <xs:sequence>
    <xs:element name="offsetDistance" type="D2LogicalModel:MetresAsNonNegativeInteger" minOccurs="1" maxOccurs="1"/>
    <xs:element name="offsetDistanceExtension" type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>

```

[top](#)

### Complex Type: OpenlrBaseLocationReferencePoint

Super-types: None  
Sub-types:

- [OpenlrLastLocationReferencePoint](#) (by extension)
- [OpenlrLocationReferencePoint](#) (by extension)

**Name** OpenlrBaseLocationReferencePoint  
**Abstract** yes  
**Documentation** Base class used to hold data about a reference point.

#### XML Instance Representation

```

<...>
<D2LogicalModel:openlrCoordinate> D2LogicalModel:PointCoordinates </D2LogicalModel:openlrCoordinate> [1]
<D2LogicalModel:openlrLineAttributes> D2LogicalModel:OpenlrLineAttributes </D2LogicalModel:openlrLineAttributes>
[1]
<D2LogicalModel:openlrBaseLocationReferencePointExtension> D2LogicalModel:_ExtensionType
</D2LogicalModel:openlrBaseLocationReferencePointExtension> [0..1]
</...>

```

#### Schema Component Representation

```

<xs:complexType name="OpenlrBaseLocationReferencePoint" abstract="true">
  <xs:sequence>
    <xs:element name="openlrCoordinate" type="D2LogicalModel:PointCoordinates"/>
    <xs:element name="openlrLineAttributes" type="D2LogicalModel:OpenlrLineAttributes"/>
    <xs:element name="openlrBaseLocationReferencePointExtension" type="D2LogicalModel:_ExtensionType"
minOccurs="0"/>
  </xs:sequence>
</xs:complexType>

```

[top](#)

### Complex Type: OpenlrBasePointLocation

Super-types:	None
Sub-types:	<ul style="list-style-type: none"> <li>• <a href="#">OpenlrPointAlongLine</a> (by extension)</li> <li>• <a href="#">OpenlrPoiWithAccessPoint</a> (by extension)</li> </ul>

<b>Name</b>	OpenlrBasePointLocation
<b>Abstract</b>	yes
<b>Documentation</b>	Holds common data that are used both in OpenlrPointAccessPoint and OpenlrPointAlongLine.

#### XML Instance Representation

```
<...>
  <D2LogicalModel:openlrSideOfRoad> D2LogicalModel:OpenlrSideOfRoadEnum </D2LogicalModel:openlrSideOfRoad> [1] ?
  <D2LogicalModel:openlrOrientation> D2LogicalModel:OpenlrOrientationEnum </D2LogicalModel:openlrOrientation> [1] ?
  <D2LogicalModel:openlrPositiveOffset> D2LogicalModel:MetresAsNonNegativeInteger
</D2LogicalModel:openlrPositiveOffset> [0..1] ?
  <D2LogicalModel:openlrLocationReferencePoint> D2LogicalModel:OpenlrLocationReferencePoint
</D2LogicalModel:openlrLocationReferencePoint> [1]
  <D2LogicalModel:openlrLastLocationReferencePoint> D2LogicalModel:OpenlrLastLocationReferencePoint
</D2LogicalModel:openlrLastLocationReferencePoint> [1]
  <D2LogicalModel:openlrBasePointLocationExtension> D2LogicalModel:_ExtensionType
</D2LogicalModel:openlrBasePointLocationExtension> [0..1]
</...>
```

#### Schema Component Representation

```
<xs:complexType name="OpenlrBasePointLocation" abstract="true">
  <xs:sequence>
    <xs:element name="openlrSideOfRoad" type="D2LogicalModel:OpenlrSideOfRoadEnum" minOccurs="1" maxOccurs="1"/>
    <xs:element name="openlrOrientation" type="D2LogicalModel:OpenlrOrientationEnum" minOccurs="1" maxOccurs="1"/>
    <xs:element name="openlrPositiveOffset" type="D2LogicalModel:MetresAsNonNegativeInteger" minOccurs="0"
maxOccurs="1"/>
    <xs:element name="openlrLocationReferencePoint" type="D2LogicalModel:OpenlrLocationReferencePoint"/>
    <xs:element name="openlrLastLocationReferencePoint" type="D2LogicalModel:OpenlrLastLocationReferencePoint"/>
    <xs:element name="openlrBasePointLocationExtension" type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
```

[top](#)

### Complex Type: [OpenlrExtendedLinear](#)

Super-types:	None
Sub-types:	None

<b>Name</b>	OpenlrExtendedLinear
<b>Abstract</b>	no
<b>Documentation</b>	Extension class for OpenLR Line location reference

#### XML Instance Representation

```
<...>
  <D2LogicalModel:firstDirection> D2LogicalModel:OpenlrLineLocationReference </D2LogicalModel:firstDirection> [1] ?
  <D2LogicalModel:oppositeDirection> D2LogicalModel:OpenlrLineLocationReference </D2LogicalModel:oppositeDirection>
[0..1] ?
</...>
```

#### Schema Component Representation

```
<xs:complexType name="OpenlrExtendedLinear">
  <xs:sequence>
    <xs:element name="firstDirection" type="D2LogicalModel:OpenlrLineLocationReference"/>
    <xs:element name="oppositeDirection" type="D2LogicalModel:OpenlrLineLocationReference" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
```

[top](#)

### Complex Type: [OpenlrExtendedPoint](#)

Super-types:	None
Sub-types:	None

<b>Name</b>	OpenlrExtendedPoint
<b>Abstract</b>	no
<b>Documentation</b>	Extension class for OpenLR point.

#### XML Instance Representation

```
<...>
  <D2LogicalModel:openlrPointLocationReference> D2LogicalModel:OpenlrPointLocationReference
</D2LogicalModel:openlrPointLocationReference> [1]
</...>
```

#### Schema Component Representation

```
<xs:complexType name="OpenlrExtendedPoint">
  <xs:sequence>
    <xs:element name="openlrPointLocationReference" type="D2LogicalModel:OpenlrPointLocationReference"/>
  </xs:sequence>
</xs:complexType>
```

[top](#)

## Complex Type: **OpenlrGeoCoordinate**

Super-types: None  
Sub-types: None

**Name** OpenlrGeoCoordinate  
**Abstract** no  
**Documentation** A geo-coordinate pair is a position in a map defined by its longitude and latitude coordinate values.

### XML Instance Representation

```
<...>  
<D2LogicalModel:openlrCoordinate> D2LogicalModel:PointCoordinates </D2LogicalModel:openlrCoordinate> [1]  
<D2LogicalModel:openlrGeoCoordinateExtension> D2LogicalModel:_ExtensionType  
</D2LogicalModel:openlrGeoCoordinateExtension> [0..1]  
</...>
```

### Schema Component Representation

```
<xs:complexType name="OpenlrGeoCoordinate">  
  <xs:sequence>  
    <xs:element name="openlrCoordinate" type="D2LogicalModel:PointCoordinates"/>  
    <xs:element name="openlrGeoCoordinateExtension" type="D2LogicalModel:_ExtensionType" minOccurs="0"/>  
  </xs:sequence>  
</xs:complexType>
```

[top](#)

## Complex Type: **OpenlrLastLocationReferencePoint**

Super-types: [OpenlrBaseLocationReferencePoint](#) < **OpenlrLastLocationReferencePoint** (by extension)  
Sub-types: None

**Name** OpenlrLastLocationReferencePoint  
**Abstract** no  
**Documentation** The sequence of location reference points is terminated by a last location reference point.

### XML Instance Representation

```
<...>  
<D2LogicalModel:openlrCoordinate> D2LogicalModel:PointCoordinates </D2LogicalModel:openlrCoordinate> [1]  
<D2LogicalModel:openlrLineAttributes> D2LogicalModel:OpenlrLineAttributes </D2LogicalModel:openlrLineAttributes>  
[1]  
<D2LogicalModel:openlrBaseLocationReferencePointExtension> D2LogicalModel:_ExtensionType  
</D2LogicalModel:openlrBaseLocationReferencePointExtension> [0..1]  
<D2LogicalModel:openlrLastLocationReferencePointExtension> D2LogicalModel:_ExtensionType  
</D2LogicalModel:openlrLastLocationReferencePointExtension> [0..1]  
</...>
```

### Schema Component Representation

```
<xs:complexType name="OpenlrLastLocationReferencePoint">  
  <xs:complexContent>  
    <xs:extension base="D2LogicalModel:OpenlrBaseLocationReferencePoint">  
      <xs:sequence>  
        <xs:element name="openlrLastLocationReferencePointExtension" type="D2LogicalModel:_ExtensionType"  
          minOccurs="0"/>  
      </xs:sequence>  
    </xs:extension>  
  </xs:complexContent>  
</xs:complexType>
```

[top](#)

## Complex Type: **OpenlrLineAttributes**

Super-types: None  
Sub-types: None

**Name** OpenlrLineAttributes  
**Abstract** no  
**Documentation** Line attributes are part of a location reference point and consists of functional road class (FRC), form of way (FOW) and bearing (BEAR) data.

### XML Instance Representation

```
<...>  
<D2LogicalModel:openlrFunctionalRoadClass> D2LogicalModel:OpenlrFunctionalRoadClassEnum  
</D2LogicalModel:openlrFunctionalRoadClass> [1] ?  
<D2LogicalModel:openlrFormOfWay> D2LogicalModel:OpenlrFormOfWayEnum </D2LogicalModel:openlrFormOfWay> [1] ?  
<D2LogicalModel:openlrBearing> D2LogicalModel:AngleInDegrees </D2LogicalModel:openlrBearing> [1] ?  
<D2LogicalModel:openlrLineAttributesExtension> D2LogicalModel:_ExtensionType  
</D2LogicalModel:openlrLineAttributesExtension> [0..1]  
</...>
```

### Schema Component Representation

```
<xs:complexType name="OpenlrLineAttributes">  
  <xs:sequence>  
    <xs:element name="openlrFunctionalRoadClass" type="D2LogicalModel:OpenlrFunctionalRoadClassEnum" minOccurs="1"  
      maxOccurs="1"/>  
  </xs:sequence>  
</xs:complexType>
```



```

<xs:element name="openlrFormOfWay" type="D2LogicalModel:OpenlrFormOfWayEnum" minOccurs="1" maxOccurs="1"/>
<xs:element name="openlrBearing" type="D2LogicalModel:AngleInDegrees" minOccurs="1" maxOccurs="1"/>
<xs:element name="openlrLineAttributesExtension" type="D2LogicalModel: _ExtensionType" minOccurs="0"/>
</xs:sequence>
</xs:complexType>

```

[top](#)

## Complex Type: OpenlrLineLocationReference

Super-types:	None
Sub-types:	None

<b>Name</b>	OpenlrLineLocationReference
<b>Abstract</b>	no
<b>Documentation</b>	A LineLocationReference is defined by an ordered sequence of location reference points and a terminating last location reference point.

### XML Instance Representation

```

<...>
  <D2LogicalModel:openlrLocationReferencePoint> D2LogicalModel:OpenlrLocationReferencePoint
</D2LogicalModel:openlrLocationReferencePoint> [1..*]
  <D2LogicalModel:openlrLastLocationReferencePoint> D2LogicalModel:OpenlrLastLocationReferencePoint
</D2LogicalModel:openlrLastLocationReferencePoint> [1]
  <D2LogicalModel:openlrOffsets> D2LogicalModel:OpenlrOffsets </D2LogicalModel:openlrOffsets> [0..1]
  <D2LogicalModel:openlrLineLocationReferenceExtension> D2LogicalModel: _ExtensionType
</D2LogicalModel:openlrLineLocationReferenceExtension> [0..1]
</...>

```

### Schema Component Representation

```

<xs:complexType name="OpenlrLineLocationReference">
  <xs:sequence>
    <xs:element name="openlrLocationReferencePoint" type="D2LogicalModel:OpenlrLocationReferencePoint"
      maxOccurs="unbounded"/>
    <xs:element name="openlrLastLocationReferencePoint" type="D2LogicalModel:OpenlrLastLocationReferencePoint"/>
    <xs:element name="openlrOffsets" type="D2LogicalModel:OpenlrOffsets" minOccurs="0"/>
    <xs:element name="openlrLineLocationReferenceExtension" type="D2LogicalModel: _ExtensionType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>

```

[top](#)

## Complex Type: OpenlrLocationReferencePoint

Super-types:	<a href="#">OpenlrBaseLocationReferencePoint</a> < OpenlrLocationReferencePoint (by extension)
Sub-types:	None

<b>Name</b>	OpenlrLocationReferencePoint
<b>Abstract</b>	no
<b>Documentation</b>	The basis of a location reference is a sequence of location reference points (LRPs).

### XML Instance Representation

```

<...>
  <D2LogicalModel:openlrCoordinate> D2LogicalModel:PointCoordinates </D2LogicalModel:openlrCoordinate> [1]
  <D2LogicalModel:openlrLineAttributes> D2LogicalModel:OpenlrLineAttributes </D2LogicalModel:openlrLineAttributes>
  [1]
  <D2LogicalModel:openlrBaseLocationReferencePointExtension> D2LogicalModel: _ExtensionType
</D2LogicalModel:openlrBaseLocationReferencePointExtension> [0..1]
  <D2LogicalModel:openlrPathAttributes> D2LogicalModel:OpenlrPathAttributes </D2LogicalModel:openlrPathAttributes>
  [1]
  <D2LogicalModel:openlrLocationReferencePointExtension> D2LogicalModel: _ExtensionType
</D2LogicalModel:openlrLocationReferencePointExtension> [0..1]
</...>

```

### Schema Component Representation

```

<xs:complexType name="OpenlrLocationReferencePoint">
  <xs:complexContent>
    <xs:extension base="D2LogicalModel:OpenlrBaseLocationReferencePoint">
      <xs:sequence>
        <xs:element name="openlrPathAttributes" type="D2LogicalModel:OpenlrPathAttributes"/>
        <xs:element name="openlrLocationReferencePointExtension" type="D2LogicalModel: _ExtensionType"
          minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

```

[top](#)

## Complex Type: OpenlrOffsets

Super-types:	None
Sub-types:	None

<b>Name</b>	OpenlrOffsets
<b>Abstract</b>	no
<b>Documentation</b>	Offsets are used to locate the start and end of a location more precisely than bounding to the nodes in a



network.

#### XML Instance Representation

```
<...>
  <D2LogicalModel:openlrPositiveOffset> D2LogicalModel:MetresAsNonNegativeInteger
</D2LogicalModel:openlrPositiveOffset> [0..1] ?
  <D2LogicalModel:openlrNegativeOffset> D2LogicalModel:MetresAsNonNegativeInteger
</D2LogicalModel:openlrNegativeOffset> [0..1] ?
  <D2LogicalModel:openlrOffsetsExtension> D2LogicalModel: _ExtensionType </D2LogicalModel:openlrOffsetsExtension>
[0..1]
</...>
```

#### Schema Component Representation

```
<xs:complexType name="OpenlrOffsets">
  <xs:sequence>
    <xs:element name="openlrPositiveOffset" type="D2LogicalModel:MetresAsNonNegativeInteger" minOccurs="0"
maxOccurs="1"/>
    <xs:element name="openlrNegativeOffset" type="D2LogicalModel:MetresAsNonNegativeInteger" minOccurs="0"
maxOccurs="1"/>
    <xs:element name="openlrOffsetsExtension" type="D2LogicalModel: _ExtensionType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
```

[top](#)

### Complex Type: **OpenlrPathAttributes**

Super-types:	None
Sub-types:	None

**Name** OpenlrPathAttributes

**Abstract** no

**Documentation** The field path attributes is part of a location reference point (except for the last location reference point) and consists of lowest functional road class (LFRCNP) and distance to next point (DNP) data.

#### XML Instance Representation

```
<...>
  <D2LogicalModel:openlrLowestFRCToNextLRPoint> D2LogicalModel:OpenlrFunctionalRoadClassEnum
</D2LogicalModel:openlrLowestFRCToNextLRPoint> [1] ?
  <D2LogicalModel:openlrDistanceToNextLRPoint> D2LogicalModel:NonNegativeInteger
</D2LogicalModel:openlrDistanceToNextLRPoint> [1] ?
  <D2LogicalModel:openlrPathAttributesExtension> D2LogicalModel: _ExtensionType
</D2LogicalModel:openlrPathAttributesExtension> [0..1]
</...>
```

#### Schema Component Representation

```
<xs:complexType name="OpenlrPathAttributes">
  <xs:sequence>
    <xs:element name="openlrLowestFRCToNextLRPoint" type="D2LogicalModel:OpenlrFunctionalRoadClassEnum"
minOccurs="1" maxOccurs="1"/>
    <xs:element name="openlrDistanceToNextLRPoint" type="D2LogicalModel:NonNegativeInteger" minOccurs="1"
maxOccurs="1"/>
    <xs:element name="openlrPathAttributesExtension" type="D2LogicalModel: _ExtensionType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
```

[top](#)

### Complex Type: **OpenlrPoiWithAccessPoint**

Super-types:	<a href="#">OpenlrBasePointLocation</a> < <a href="#">OpenlrPoiWithAccessPoint</a> (by extension)
Sub-types:	None

**Name** OpenlrPoiWithAccessPoint

**Abstract** no

**Documentation** Point along line with access is a point location which is defined by a line,an offset value and a coordinate.

#### XML Instance Representation

```
<...>
  <D2LogicalModel:openlrSideOfRoad> D2LogicalModel:OpenlrSideOfRoadEnum </D2LogicalModel:openlrSideOfRoad> [1] ?
  <D2LogicalModel:openlrOrientation> D2LogicalModel:OpenlrOrientationEnum </D2LogicalModel:openlrOrientation> [1] ?
  <D2LogicalModel:openlrPositiveOffset> D2LogicalModel:MetresAsNonNegativeInteger
</D2LogicalModel:openlrPositiveOffset> [0..1] ?
  <D2LogicalModel:openlrLocationReferencePoint> D2LogicalModel:OpenlrLocationReferencePoint
</D2LogicalModel:openlrLocationReferencePoint> [1]
  <D2LogicalModel:openlrLastLocationReferencePoint> D2LogicalModel:OpenlrLastLocationReferencePoint
</D2LogicalModel:openlrLastLocationReferencePoint> [1]
  <D2LogicalModel:openlrBasePointLocationExtension> D2LogicalModel: _ExtensionType
</D2LogicalModel:openlrBasePointLocationExtension> [0..1]
  <D2LogicalModel:openlrCoordinate> D2LogicalModel:PointCoordinates </D2LogicalModel:openlrCoordinate> [1] ?
  <D2LogicalModel:openlrPoiWithAccessPointExtension> D2LogicalModel: _ExtensionType
</D2LogicalModel:openlrPoiWithAccessPointExtension> [0..1]
</...>
```

#### Schema Component Representation

```
<xs:complexType name="OpenlrPoiWithAccessPoint">
  <xs:complexContent>
    <xs:extension base="D2LogicalModel:OpenlrBasePointLocation">
      <xs:sequence>
```

```

    <xs:element name="openlrCoordinate" type="D2LogicalModel:PointCoordinates"/>
    <xs:element name="openlrPoiWithAccessPointExtension" type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
  </xs:sequence>
</xs:extension>
</xs:complexContent>
</xs:complexType>

```

[top](#)

## Complex Type: **OpenlrPointAlongLine**

Super-types: [OpenlrBasePointLocation](#) < **OpenlrPointAlongLine** (by extension)  
 Sub-types: None

**Name** OpenlrPointAlongLine  
**Abstract** no  
**Documentation** Point along a line

### XML Instance Representation

```

<...>
  <D2LogicalModel:openlrSideOfRoad D2LogicalModel:OpenlrSideOfRoadEnum </D2LogicalModel:openlrSideOfRoad> [1] ?
  <D2LogicalModel:openlrOrientation D2LogicalModel:OpenlrOrientationEnum </D2LogicalModel:openlrOrientation> [1] ?
  <D2LogicalModel:openlrPositiveOffset D2LogicalModel:MetresAsNonNegativeInteger
  </D2LogicalModel:openlrPositiveOffset> [0..1] ?
  <D2LogicalModel:openlrLocationReferencePoint D2LogicalModel:OpenlrLocationReferencePoint
  </D2LogicalModel:openlrLocationReferencePoint> [1]
  <D2LogicalModel:openlrLastLocationReferencePoint D2LogicalModel:OpenlrLastLocationReferencePoint
  </D2LogicalModel:openlrLastLocationReferencePoint> [1]
  <D2LogicalModel:openlrBasePointLocationExtension D2LogicalModel:_ExtensionType
  </D2LogicalModel:openlrBasePointLocationExtension> [0..1]
  <D2LogicalModel:openlrPointAlongLineExtension D2LogicalModel:_ExtensionType
  </D2LogicalModel:openlrPointAlongLineExtension> [0..1]
</...>

```

### Schema Component Representation

```

<xs:complexType name="OpenlrPointAlongLine">
  <xs:complexContent>
    <xs:extension base="D2LogicalModel:OpenlrBasePointLocation">
      <xs:sequence>
        <xs:element name="openlrPointAlongLineExtension" type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

```

[top](#)

## Complex Type: **OpenlrPointLocationReference**

Super-types: None  
 Sub-types: None

**Name** OpenlrPointLocationReference  
**Abstract** no  
**Documentation** A point location is a zero-dimensional element in a map that specifies a geometric location.

### XML Instance Representation

```

<...>
  <D2LogicalModel:openlrGeoCoordinate D2LogicalModel:OpenlrGeoCoordinate </D2LogicalModel:openlrGeoCoordinate>
  [0..1]
  <D2LogicalModel:openlrPoiWithAccessPoint D2LogicalModel:OpenlrPoiWithAccessPoint
  </D2LogicalModel:openlrPoiWithAccessPoint> [0..1]
  <D2LogicalModel:openlrPointAlongLine D2LogicalModel:OpenlrPointAlongLine </D2LogicalModel:openlrPointAlongLine>
  [0..1]
  <D2LogicalModel:openlrPointLocationReferenceExtension D2LogicalModel:_ExtensionType
  </D2LogicalModel:openlrPointLocationReferenceExtension> [0..1]
</...>

```

### Schema Component Representation

```

<xs:complexType name="OpenlrPointLocationReference">
  <xs:sequence>
    <xs:element name="openlrGeoCoordinate" type="D2LogicalModel:OpenlrGeoCoordinate" minOccurs="0"/>
    <xs:element name="openlrPoiWithAccessPoint" type="D2LogicalModel:OpenlrPoiWithAccessPoint" minOccurs="0"/>
    <xs:element name="openlrPointAlongLine" type="D2LogicalModel:OpenlrPointAlongLine" minOccurs="0"/>
    <xs:element name="openlrPointLocationReferenceExtension" type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>

```

[top](#)

## Complex Type: **PayloadPublication**

Super-types: None  
 Sub-types:
 

- [VmsPublication](#) (by extension)

**Name** PayloadPublication  
**Abstract** yes

## XML Instance Representation

```
<...
  lang="D2LogicalModel:Language [1] ?">
  <D2LogicalModel:publicationTime> D2LogicalModel:DateTime </D2LogicalModel:publicationTime> [1] ?
  <D2LogicalModel:publicationCreator> D2LogicalModel:InternationalIdentifier </D2LogicalModel:publicationCreator>
  [1]
  <D2LogicalModel:payloadPublicationExtension> D2LogicalModel:_ExtensionType
  </D2LogicalModel:payloadPublicationExtension> [0..1]
</...>
```

## Schema Component Representation

```
<xs:complexType name="PayloadPublication" abstract="true">
  <xs:sequence>
    <xs:element name="publicationTime" type="D2LogicalModel:DateTime" minOccurs="1" maxOccurs="1"/>
    <xs:element name="publicationCreator" type="D2LogicalModel:InternationalIdentifier"/>
    <xs:element name="payloadPublicationExtension" type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
  </xs:sequence>
  <xs:attribute name="lang" type="D2LogicalModel:Language" use="required"/>
</xs:complexType>
```

[top](#)

## Complex Type: PcuFlowValue

Super-types: [DataValue](#) < PcuFlowValue (by extension)

Sub-types: None

**Name** PcuFlowValue  
**Abstract** no  
**Documentation** A measured or calculated value of the flow rate of passenger car units.

## XML Instance Representation

```
<...
  accuracy="D2LogicalModel:Percentage [0..1] ?"
  computationalMethod="D2LogicalModel:ComputationMethodEnum [0..1] ?"
  numberOfIncompleteInputs="D2LogicalModel:NonNegativeInteger [0..1] ?"
  numberOfInputValuesUsed="D2LogicalModel:NonNegativeInteger [0..1] ?"
  smoothingFactor="D2LogicalModel:Float [0..1] ?"
  standardDeviation="D2LogicalModel:Float [0..1] ?"
  supplierCalculatedDataQuality="D2LogicalModel:Percentage [0..1] ?">
  <D2LogicalModel:dataError> D2LogicalModel:Boolean </D2LogicalModel:dataError> [0..1] ?
  <D2LogicalModel:reasonForDataError> D2LogicalModel:MultilingualString </D2LogicalModel:reasonForDataError> [0..1]
  ?
  <D2LogicalModel:dataValueExtension> D2LogicalModel:_ExtensionType </D2LogicalModel:dataValueExtension> [0..1]
  <D2LogicalModel:pcuFlowRate> D2LogicalModel:PassengerCarUnitsPerHour </D2LogicalModel:pcuFlowRate> [1] ?
  <D2LogicalModel:pcuFlowValueExtension> D2LogicalModel:_ExtensionType </D2LogicalModel:pcuFlowValueExtension>
  [0..1]
</...>
```

## Schema Component Representation

```
<xs:complexType name="PcuFlowValue">
  <xs:complexContent>
    <xs:extension base="D2LogicalModel:DataValue">
      <xs:sequence>
        <xs:element name="pcuFlowRate" type="D2LogicalModel:PassengerCarUnitsPerHour" minOccurs="1" maxOccurs="1"/>
        <xs:element name="pcuFlowValueExtension" type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

[top](#)

## Complex Type: PercentageDistanceAlongLinearElement

Super-types: [DistanceAlongLinearElement](#) < PercentageDistanceAlongLinearElement (by extension)

Sub-types: None

**Name** PercentageDistanceAlongLinearElement  
**Abstract** no  
**Documentation** Distance of a point along a linear element measured from the start node expressed as a percentage of the whole length of the linear element, where start node is relative to the element definition rather than the direction of traffic flow.

## XML Instance Representation

```
<...>
  <D2LogicalModel:distanceAlongLinearElementExtension> D2LogicalModel:_ExtensionType
  </D2LogicalModel:distanceAlongLinearElementExtension> [0..1]
  <D2LogicalModel:percentageDistanceAlong> D2LogicalModel:Percentage </D2LogicalModel:percentageDistanceAlong> [1] ?
  <D2LogicalModel:percentageDistanceAlongLinearElementExtension> D2LogicalModel:_ExtensionType
  </D2LogicalModel:percentageDistanceAlongLinearElementExtension> [0..1]
</...>
```

## Schema Component Representation

```
<xs:complexType name="PercentageDistanceAlongLinearElement">
```

```

<xs:complexContent>
  <xs:extension base="D2LogicalModel:DistanceAlongLinearElement">
    <xs:sequence>
      <xs:element name="percentageDistanceAlong" type="D2LogicalModel:Percentage" minOccurs="1" maxOccurs="1"/>
      <xs:element name="percentageDistanceAlongLinearElementExtension" type="D2LogicalModel:_ExtensionType"
        minOccurs="0"/>
    </xs:sequence>
  </xs:extension>
</xs:complexContent>
</xs:complexType>

```

[top](#)

## Complex Type: PictogramDisplayAreaSettings

Super-types: None  
Sub-types: None

**Name** PictogramDisplayAreaSettings  
**Abstract** no  
**Documentation** Settings specific to a distinct pictogram display area on the VMS (where pictogramIndex indicates which pictogram area it relates to if there is more than one pictogram display area on the VMS).

### XML Instance Representation

```

<...>
  <D2LogicalModel:pictogramLanternsOn> D2LogicalModel:Boolean </D2LogicalModel:pictogramLanternsOn> [0..1] ?
  <D2LogicalModel:pictogramLuminanceOverride> D2LogicalModel:Boolean </D2LogicalModel:pictogramLuminanceOverride>
  [0..1] ?
  <D2LogicalModel:pictogramLuminanceLevel> D2LogicalModel:NonNegativeInteger
  </D2LogicalModel:pictogramLuminanceLevel> [0..1] ?
  <D2LogicalModel:pictogramLuminanceLevelName> D2LogicalModel:VmsLuminanceLevelEnum
  </D2LogicalModel:pictogramLuminanceLevelName> [0..1] ?
  <D2LogicalModel:pictogramDisplayAreaSettingsExtension> D2LogicalModel:_ExtensionType
  </D2LogicalModel:pictogramDisplayAreaSettingsExtension> [0..1]
</...>

```

### Schema Component Representation

```

<xs:complexType name="PictogramDisplayAreaSettings">
  <xs:sequence>
    <xs:element name="pictogramLanternsOn" type="D2LogicalModel:Boolean" minOccurs="0" maxOccurs="1"/>
    <xs:element name="pictogramLuminanceOverride" type="D2LogicalModel:Boolean" minOccurs="0" maxOccurs="1"/>
    <xs:element name="pictogramLuminanceLevel" type="D2LogicalModel:NonNegativeInteger" minOccurs="0"
      maxOccurs="1"/>
    <xs:element name="pictogramLuminanceLevelName" type="D2LogicalModel:VmsLuminanceLevelEnum" minOccurs="0"
      maxOccurs="1"/>
    <xs:element name="pictogramDisplayAreaSettingsExtension" type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>

```

[top](#)

## Complex Type: Point

Super-types: [GroupOfLocations](#) < [Location](#) (by extension) < [NetworkLocation](#) (by extension) < [Point](#) (by extension)  
Sub-types: None

**Name** Point  
**Abstract** no  
**Documentation** A single geospatial point.

### XML Instance Representation

```

<...>
  <D2LogicalModel:groupOfLocationsExtension> D2LogicalModel:_ExtensionType
  </D2LogicalModel:groupOfLocationsExtension> [0..1]
  <D2LogicalModel:locationForDisplay> D2LogicalModel:PointCoordinates </D2LogicalModel:locationForDisplay> [0..1] ?
  <D2LogicalModel:locationExtension> D2LogicalModel:_ExtensionType </D2LogicalModel:locationExtension> [0..1]
  <D2LogicalModel:supplementaryPositionalDescription> D2LogicalModel:SupplementaryPositionalDescription
  </D2LogicalModel:supplementaryPositionalDescription> [0..1]
  <D2LogicalModel:networkLocationExtension> D2LogicalModel:_ExtensionType </D2LogicalModel:networkLocationExtension>
  [0..1]
  <D2LogicalModel:tpegPointLocation> D2LogicalModel:TpegPointLocation </D2LogicalModel:tpegPointLocation> [0..1]
  <D2LogicalModel>alertCPoint> D2LogicalModel:AlertCPoint </D2LogicalModel>alertCPoint> [0..1]
  <D2LogicalModel:pointAlongLinearElement> D2LogicalModel:PointAlongLinearElement
  </D2LogicalModel:pointAlongLinearElement> [0..1]
  <D2LogicalModel:pointByCoordinates> D2LogicalModel:PointByCoordinates </D2LogicalModel:pointByCoordinates> [0..1]
  <D2LogicalModel:pointExtension> D2LogicalModel:_PointExtensionType </D2LogicalModel:pointExtension> [0..1]
</...>

```

### Schema Component Representation

```

<xs:complexType name="Point">
  <xs:complexContent>
    <xs:extension base="D2LogicalModel:NetworkLocation">
      <xs:sequence>
        <xs:element name="tpegPointLocation" type="D2LogicalModel:TpegPointLocation" minOccurs="0"/>
        <xs:element name="alertCPoint" type="D2LogicalModel:AlertCPoint" minOccurs="0"/>
        <xs:element name="pointAlongLinearElement" type="D2LogicalModel:PointAlongLinearElement" minOccurs="0"/>
        <xs:element name="pointByCoordinates" type="D2LogicalModel:PointByCoordinates" minOccurs="0"/>
        <xs:element name="pointExtension" type="D2LogicalModel:_PointExtensionType" minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

```

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

[top](#)

## Complex Type: **PointAlongLinearElement**

Super-types: None  
Sub-types: None

**Name** PointAlongLinearElement  
**Abstract** no  
**Documentation** A point on a linear element where the linear element is either a part of or the whole of a linear object (i.e. a road), consistent with ISO 19148 definitions.

### XML Instance Representation

```
<...>  
<D2LogicalModel:administrativeAreaOfPoint> D2LogicalModel:MultilingualString  
</D2LogicalModel:administrativeAreaOfPoint> [0..1] ?  
<D2LogicalModel:directionBoundAtPoint> D2LogicalModel:DirectionEnum </D2LogicalModel:directionBoundAtPoint> [0..1]  
 ?  
<D2LogicalModel:directionRelativeAtPoint> D2LogicalModel:LinearReferencingDirectionEnum  
</D2LogicalModel:directionRelativeAtPoint> [0..1] ?  
<D2LogicalModel:heightGradeOfPoint> D2LogicalModel:HeightGradeEnum </D2LogicalModel:heightGradeOfPoint> [0..1] ?  
<D2LogicalModel:linearElement> D2LogicalModel:LinearElement </D2LogicalModel:linearElement> [1]  
<D2LogicalModel:distanceAlongLinearElement> D2LogicalModel:DistanceAlongLinearElement  
</D2LogicalModel:distanceAlongLinearElement> [1]  
<D2LogicalModel:pointAlongLinearElementExtension> D2LogicalModel:_ExtensionType  
</D2LogicalModel:pointAlongLinearElementExtension> [0..1]  
</...>
```

### Schema Component Representation

```
<xs:complexType name="PointAlongLinearElement">  
  <xs:sequence>  
    <xs:element name="administrativeAreaOfPoint" type="D2LogicalModel:MultilingualString" minOccurs="0"  
      maxOccurs="1"/>  
    <xs:element name="directionBoundAtPoint" type="D2LogicalModel:DirectionEnum" minOccurs="0" maxOccurs="1"/>  
    <xs:element name="directionRelativeAtPoint" type="D2LogicalModel:LinearReferencingDirectionEnum" minOccurs="0"  
      maxOccurs="1"/>  
    <xs:element name="heightGradeOfPoint" type="D2LogicalModel:HeightGradeEnum" minOccurs="0" maxOccurs="1"/>  
    <xs:element name="linearElement" type="D2LogicalModel:LinearElement"/>  
    <xs:element name="distanceAlongLinearElement" type="D2LogicalModel:DistanceAlongLinearElement"/>  
    <xs:element name="pointAlongLinearElementExtension" type="D2LogicalModel:_ExtensionType" minOccurs="0"/>  
  </xs:sequence>  
</xs:complexType>
```

[top](#)

## Complex Type: **PointByCoordinates**

Super-types: None  
Sub-types: None

**Name** PointByCoordinates  
**Abstract** no  
**Documentation** A single point defined only by a coordinate set with an optional bearing direction.

### XML Instance Representation

```
<...>  
<D2LogicalModel:bearing> D2LogicalModel:NonNegativeInteger </D2LogicalModel:bearing> [0..1] ?  
<D2LogicalModel:pointCoordinates> D2LogicalModel:PointCoordinates </D2LogicalModel:pointCoordinates> [1]  
<D2LogicalModel:pointByCoordinatesExtension> D2LogicalModel:_ExtensionType  
</D2LogicalModel:pointByCoordinatesExtension> [0..1]  
</...>
```

### Schema Component Representation

```
<xs:complexType name="PointByCoordinates">  
  <xs:sequence>  
    <xs:element name="bearing" type="D2LogicalModel:NonNegativeInteger" minOccurs="0" maxOccurs="1"/>  
    <xs:element name="pointCoordinates" type="D2LogicalModel:PointCoordinates"/>  
    <xs:element name="pointByCoordinatesExtension" type="D2LogicalModel:_ExtensionType" minOccurs="0"/>  
  </xs:sequence>  
</xs:complexType>
```

[top](#)

## Complex Type: **PointCoordinates**

Super-types: None  
Sub-types: None

**Name** PointCoordinates  
**Abstract** no  
**Documentation** A pair of coordinates defining the geodetic position of a single point using the European Terrestrial Reference System 1989 (ETRS89).

### XML Instance Representation

```

<...>
  <D2LogicalModel:latitude> D2LogicalModel:Float </D2LogicalModel:latitude> [1] ?
  <D2LogicalModel:longitude> D2LogicalModel:Float </D2LogicalModel:longitude> [1] ?
  <D2LogicalModel:pointCoordinatesExtension> D2LogicalModel:_ExtensionType
  </D2LogicalModel:pointCoordinatesExtension> [0..1]
</...>

```

#### Schema Component Representation

```

<xs:complexType name="PointCoordinates">
  <xs:sequence>
    <xs:element name="latitude" type="D2LogicalModel:Float" minOccurs="1" maxOccurs="1"/>
    <xs:element name="longitude" type="D2LogicalModel:Float" minOccurs="1" maxOccurs="1"/>
    <xs:element name="pointCoordinatesExtension" type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>

```

[top](#)

### Complex Type: PointExtended

Super-types:	None
Sub-types:	None

<b>Name</b>	PointExtended
<b>Abstract</b>	no
<b>Documentation</b>	Extension point for 'Point' to support the description of junctions (and other alternative point descriptions).

#### XML Instance Representation

```

<...>
  <D2LogicalModel:description> D2LogicalModel:MultilingualString </D2LogicalModel:description> [0..1] ?
  <D2LogicalModel:junction> D2LogicalModel:Junction </D2LogicalModel:junction> [0..1]
</...>

```

#### Schema Component Representation

```

<xs:complexType name="PointExtended">
  <xs:sequence>
    <xs:element name="description" type="D2LogicalModel:MultilingualString" minOccurs="0" maxOccurs="1"/>
    <xs:element name="junction" type="D2LogicalModel:Junction" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>

```

[top](#)

### Complex Type: Referent

Super-types:	None
Sub-types:	None

<b>Name</b>	Referent
<b>Abstract</b>	no
<b>Documentation</b>	A referent on a linear object that has a known location such as a node, a reference marker (e.g. a markerpost), an intersection etc.

#### XML Instance Representation

```

<...>
  <D2LogicalModel:referentIdentifier> D2LogicalModel:String </D2LogicalModel:referentIdentifier> [1] ?
  <D2LogicalModel:referentName> D2LogicalModel:String </D2LogicalModel:referentName> [0..1] ?
  <D2LogicalModel:referentType> D2LogicalModel:ReferentTypeEnum </D2LogicalModel:referentType> [1] ?
  <D2LogicalModel:referentDescription> D2LogicalModel:MultilingualString </D2LogicalModel:referentDescription> [0..1] ?
  <D2LogicalModel:pointCoordinates> D2LogicalModel:PointCoordinates </D2LogicalModel:pointCoordinates> [0..1]
  <D2LogicalModel:referentExtension> D2LogicalModel:_ExtensionType </D2LogicalModel:referentExtension> [0..1]
</...>

```

#### Schema Component Representation

```

<xs:complexType name="Referent">
  <xs:sequence>
    <xs:element name="referentIdentifier" type="D2LogicalModel:String" minOccurs="1" maxOccurs="1"/>
    <xs:element name="referentName" type="D2LogicalModel:String" minOccurs="0" maxOccurs="1"/>
    <xs:element name="referentType" type="D2LogicalModel:ReferentTypeEnum" minOccurs="1" maxOccurs="1"/>
    <xs:element name="referentDescription" type="D2LogicalModel:MultilingualString" minOccurs="0" maxOccurs="1"/>
    <xs:element name="pointCoordinates" type="D2LogicalModel:PointCoordinates" minOccurs="0"/>
    <xs:element name="referentExtension" type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>

```

[top](#)

### Complex Type: Road

Super-types:	None
Sub-types:	<ul style="list-style-type: none"> <li>• <a href="#">RoadNode</a> (by extension)</li> </ul>

<b>Name</b>	Road
<b>Abstract</b>	no

**XML Instance Representation**

```

<...>
  <D2LogicalModel:nameOfRoad> D2LogicalModel:MultilingualString </D2LogicalModel:nameOfRoad> [0..1] ?
  <D2LogicalModel:roadIdentifier> D2LogicalModel:MultilingualString </D2LogicalModel:roadIdentifier> [0..1] ?
  <D2LogicalModel:typeOfRoad> D2LogicalModel:RoadTypeEnum </D2LogicalModel:typeOfRoad> [0..1] ?
  <D2LogicalModel:roadDestination> D2LogicalModel:MultilingualString </D2LogicalModel:roadDestination> [0..*] ?
  <D2LogicalModel:roadOrigination> D2LogicalModel:MultilingualString </D2LogicalModel:roadOrigination> [0..*] ?
  <D2LogicalModel:distanceToThisRoad> D2LogicalModel:MetresAsNonNegativeInteger </D2LogicalModel:distanceToThisRoad>
  [0..1] ?
  <D2LogicalModel:roadExtension> D2LogicalModel:_ExtensionType </D2LogicalModel:roadExtension> [0..1]
</...>

```

**Schema Component Representation**

```

<xs:complexType name="Road">
  <xs:sequence>
    <xs:element name="nameOfRoad" type="D2LogicalModel:MultilingualString" minOccurs="0" maxOccurs="1"/>
    <xs:element name="roadIdentifier" type="D2LogicalModel:MultilingualString" minOccurs="0" maxOccurs="1"/>
    <xs:element name="typeOfRoad" type="D2LogicalModel:RoadTypeEnum" minOccurs="0" maxOccurs="1"/>
    <xs:element name="roadDestination" type="D2LogicalModel:MultilingualString" minOccurs="0"
    maxOccurs="unbounded"/>
    <xs:element name="roadOrigination" type="D2LogicalModel:MultilingualString" minOccurs="0"
    maxOccurs="unbounded"/>
    <xs:element name="distanceToThisRoad" type="D2LogicalModel:MetresAsNonNegativeInteger" minOccurs="0"
    maxOccurs="1"/>
    <xs:element name="roadExtension" type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>

```

[top](#)**Complex Type: RoadNode**

Super-types: [Road](#) < RoadNode (by extension)

Sub-types: None

**Name** RoadNode

**Abstract** no

**Documentation** A road node as part of the specialised road identified by the name of a junctionon on this road.

**XML Instance Representation**

```

<...>
  <D2LogicalModel:nameOfRoad> D2LogicalModel:MultilingualString </D2LogicalModel:nameOfRoad> [0..1] ?
  <D2LogicalModel:roadIdentifier> D2LogicalModel:MultilingualString </D2LogicalModel:roadIdentifier> [0..1] ?
  <D2LogicalModel:typeOfRoad> D2LogicalModel:RoadTypeEnum </D2LogicalModel:typeOfRoad> [0..1] ?
  <D2LogicalModel:roadDestination> D2LogicalModel:MultilingualString </D2LogicalModel:roadDestination> [0..*] ?
  <D2LogicalModel:roadOrigination> D2LogicalModel:MultilingualString </D2LogicalModel:roadOrigination> [0..*] ?
  <D2LogicalModel:distanceToThisRoad> D2LogicalModel:MetresAsNonNegativeInteger </D2LogicalModel:distanceToThisRoad>
  [0..1] ?
  <D2LogicalModel:roadExtension> D2LogicalModel:_ExtensionType </D2LogicalModel:roadExtension> [0..1]
  <D2LogicalModel:junctionName> D2LogicalModel:MultilingualString </D2LogicalModel:junctionName> [1] ?
  <D2LogicalModel:roadNodeExtension> D2LogicalModel:_ExtensionType </D2LogicalModel:roadNodeExtension> [0..1]
</...>

```

**Schema Component Representation**

```

<xs:complexType name="RoadNode">
  <xs:complexContent>
    <xs:extension base="D2LogicalModel:Road">
      <xs:sequence>
        <xs:element name="junctionName" type="D2LogicalModel:MultilingualString" minOccurs="1" maxOccurs="1"/>
        <xs:element name="roadNodeExtension" type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

```

[top](#)**Complex Type: SupplementaryPositionalDescription**

Super-types: None

Sub-types: None

**Name** SupplementaryPositionalDescription

**Abstract** no

**Documentation** A collection of supplementary positional information which improves the precision of the location.

**XML Instance Representation**

```

<...
  locationPrecision="D2LogicalModel:MetresAsNonNegativeInteger [0..1] ?">
  <D2LogicalModel:locationDescriptor> D2LogicalModel:LocationDescriptorEnum </D2LogicalModel:locationDescriptor>
  [1..*] ?
  <D2LogicalModel:sequentialRampNumber> D2LogicalModel:NonNegativeInteger </D2LogicalModel:sequentialRampNumber>
  [0..1] ?
  <D2LogicalModel:affectedCarriagewayAndLanes> D2LogicalModel:AffectedCarriagewayAndLanes
  </D2LogicalModel:affectedCarriagewayAndLanes> [0..*]
  <D2LogicalModel:supplementaryPositionalDescriptionExtension> D2LogicalModel:_ExtensionType
  </D2LogicalModel:supplementaryPositionalDescriptionExtension> [0..1]

```



```
</...>
```

#### Schema Component Representation

```
<xs:complexType name="SupplementaryPositionalDescription">
  <xs:sequence>
    <xs:element name="locationDescriptor" type="D2LogicalModel:LocationDescriptorEnum" minOccurs="1"
      maxOccurs="unbounded"/>
    <xs:element name="sequentialRampNumber" type="D2LogicalModel:NonNegativeInteger" minOccurs="0" maxOccurs="1"/>
    <xs:element name="affectedCarriagewayAndLanes" type="D2LogicalModel:AffectedCarriagewayAndLanes" minOccurs="0"
      maxOccurs="unbounded"/>
    <xs:element name="supplementaryPositionalDescriptionExtension" type="D2LogicalModel:_ExtensionType"
      minOccurs="0"/>
  </xs:sequence>
  <xs:attribute name="locationPrecision" type="D2LogicalModel:MetresAsNonNegativeInteger" use="optional"/>
</xs:complexType>
```

[top](#)

### Complex Type: TextDisplayAreaSettings

Super-types:	None
Sub-types:	None

<b>Name</b>	TextDisplayAreaSettings
<b>Abstract</b>	no
<b>Documentation</b>	Settings specific to a distinct text display area on the VMS.

#### XML Instance Representation

```
<...>
  <D2LogicalModel:textLanternsOn> D2LogicalModel:Boolean </D2LogicalModel:textLanternsOn> [0..1] ?
  <D2LogicalModel:textLuminanceOverride> D2LogicalModel:Boolean </D2LogicalModel:textLuminanceOverride> [0..1] ?
  <D2LogicalModel:textLuminanceLevel> D2LogicalModel:NonNegativeInteger </D2LogicalModel:textLuminanceLevel> [0..1]
  ?
  <D2LogicalModel:textLuminanceLevelName> D2LogicalModel:VmsLuminanceLevelEnum
  </D2LogicalModel:textLuminanceLevelName> [0..1] ?
  <D2LogicalModel:textDisplayAreaSettingsExtension> D2LogicalModel:_ExtensionType
  </D2LogicalModel:textDisplayAreaSettingsExtension> [0..1]
</...>
```

#### Schema Component Representation

```
<xs:complexType name="TextDisplayAreaSettings">
  <xs:sequence>
    <xs:element name="textLanternsOn" type="D2LogicalModel:Boolean" minOccurs="0" maxOccurs="1"/>
    <xs:element name="textLuminanceOverride" type="D2LogicalModel:Boolean" minOccurs="0" maxOccurs="1"/>
    <xs:element name="textLuminanceLevel" type="D2LogicalModel:NonNegativeInteger" minOccurs="0" maxOccurs="1"/>
    <xs:element name="textLuminanceLevelName" type="D2LogicalModel:VmsLuminanceLevelEnum" minOccurs="0"
      maxOccurs="1"/>
    <xs:element name="textDisplayAreaSettingsExtension" type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
```

[top](#)

### Complex Type: TpegDescriptor

Super-types:	None
Sub-types:	<ul style="list-style-type: none"><li>• <a href="#">TpegPointDescriptor</a> (by extension)<ul style="list-style-type: none"><li>◦ <a href="#">TpegJlcPointDescriptor</a> (by extension)</li><li>◦ <a href="#">TpegJunctionPointDescriptor</a> (by extension)</li><li>◦ <a href="#">TpegOtherPointDescriptor</a> (by extension)</li></ul></li></ul>

<b>Name</b>	TpegDescriptor
<b>Abstract</b>	yes
<b>Documentation</b>	A collection of information providing descriptive references to locations using the TPEG-Loc location referencing approach.

#### XML Instance Representation

```
<...>
  <D2LogicalModel:descriptor> D2LogicalModel:MultilingualString </D2LogicalModel:descriptor> [1] ?
  <D2LogicalModel:tpegDescriptorExtension> D2LogicalModel:_ExtensionType </D2LogicalModel:tpegDescriptorExtension>
  [0..1]
</...>
```

#### Schema Component Representation

```
<xs:complexType name="TpegDescriptor" abstract="true">
  <xs:sequence>
    <xs:element name="descriptor" type="D2LogicalModel:MultilingualString" minOccurs="1" maxOccurs="1"/>
    <xs:element name="tpegDescriptorExtension" type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
```

[top](#)

### Complex Type: TpegFramedPoint

Super-types:	<a href="#">TpegPointLocation</a> < TpegFramedPoint (by extension)
Sub-types:	None



<b>Name</b>	TpegFramedPoint
<b>Abstract</b>	no
<b>Documentation</b>	A point on the road network which is framed between two other points on the same road.

#### XML Instance Representation

```

<...>
  <D2LogicalModel:tpegDirection> D2LogicalModel:DirectionEnum </D2LogicalModel:tpegDirection> [1] ?
  <D2LogicalModel:tpegPointLocationExtension> D2LogicalModel:_ExtensionType
  </D2LogicalModel:tpegPointLocationExtension> [0..1]
  <D2LogicalModel:tpegFramedPointLocationType> D2LogicalModel:TpegLoc01FramedPointLocationSubtypeEnum
  </D2LogicalModel:tpegFramedPointLocationType> [1] ?
  <D2LogicalModel:framedPoint> D2LogicalModel:TpegNonJunctionPoint </D2LogicalModel:framedPoint> [1] ?
  <D2LogicalModel:to> D2LogicalModel:TpegPoint </D2LogicalModel:to> [1] ?
  <D2LogicalModel:from> D2LogicalModel:TpegPoint </D2LogicalModel:from> [1] ?
  <D2LogicalModel:tpegFramedPointExtension> D2LogicalModel:_ExtensionType </D2LogicalModel:tpegFramedPointExtension>
  [0..1]
</...>

```

#### Schema Component Representation

```

<xs:complexType name="TpegFramedPoint">
  <xs:complexContent>
    <xs:extension base="D2LogicalModel:TpegPointLocation">
      <xs:sequence>
        <xs:element name="tpegFramedPointLocationType"
          type="D2LogicalModel:TpegLoc01FramedPointLocationSubtypeEnum" minOccurs="1" maxOccurs="1"/>
        <xs:element name="framedPoint" type="D2LogicalModel:TpegNonJunctionPoint"/>
        <xs:element name="to" type="D2LogicalModel:TpegPoint"/>
        <xs:element name="from" type="D2LogicalModel:TpegPoint"/>
        <xs:element name="tpegFramedPointExtension" type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

```

[top](#)

### Complex Type: TpegIlcPointDescriptor

<b>Super-types:</b>	<a href="#">TpegDescriptor</a> < <a href="#">TpegPointDescriptor</a> (by extension) < <a href="#">TpegIlcPointDescriptor</a> (by extension)
<b>Sub-types:</b>	None

<b>Name</b>	TpegIlcPointDescriptor
<b>Abstract</b>	no
<b>Documentation</b>	A descriptor for describing a junction by defining the intersecting roads.

#### XML Instance Representation

```

<...>
  <D2LogicalModel:descriptor> D2LogicalModel:MultilingualString </D2LogicalModel:descriptor> [1] ?
  <D2LogicalModel:tpegDescriptorExtension> D2LogicalModel:_ExtensionType </D2LogicalModel:tpegDescriptorExtension>
  [0..1]
  <D2LogicalModel:tpegPointDescriptorExtension> D2LogicalModel:_ExtensionType
  </D2LogicalModel:tpegPointDescriptorExtension> [0..1]
  <D2LogicalModel:tpegIlcPointDescriptorType> D2LogicalModel:TpegLoc03IlcPointDescriptorSubtypeEnum
  </D2LogicalModel:tpegIlcPointDescriptorType> [1] ?
  <D2LogicalModel:tpegIlcPointDescriptorExtension> D2LogicalModel:_ExtensionType
  </D2LogicalModel:tpegIlcPointDescriptorExtension> [0..1]
</...>

```

#### Schema Component Representation

```

<xs:complexType name="TpegIlcPointDescriptor">
  <xs:complexContent>
    <xs:extension base="D2LogicalModel:TpegPointDescriptor">
      <xs:sequence>
        <xs:element name="tpegIlcPointDescriptorType" type="D2LogicalModel:TpegLoc03IlcPointDescriptorSubtypeEnum"
          minOccurs="1" maxOccurs="1"/>
        <xs:element name="tpegIlcPointDescriptorExtension" type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

```

[top](#)

### Complex Type: TpegJunction

<b>Super-types:</b>	<a href="#">TpegPoint</a> < <a href="#">TpegJunction</a> (by extension)
<b>Sub-types:</b>	None

<b>Name</b>	TpegJunction
<b>Abstract</b>	no
<b>Documentation</b>	A point on the road network which is a road junction point.

#### XML Instance Representation

```

<...>
  <D2LogicalModel:tpegPointExtension> D2LogicalModel:_ExtensionType </D2LogicalModel:tpegPointExtension> [0..1]
  <D2LogicalModel:pointCoordinates> D2LogicalModel:PointCoordinates </D2LogicalModel:pointCoordinates> [1]
  <D2LogicalModel:name> D2LogicalModel:TpegJunctionPointDescriptor </D2LogicalModel:name> [0..1] ?
  <D2LogicalModel:ilc> D2LogicalModel:TpegIlcPointDescriptor </D2LogicalModel:ilc> [1..3] ?
</...>

```

```

</D2LogicalModel:otherName> D2LogicalModel:TpegOtherPointDescriptor </D2LogicalModel:otherName> [0..*] ?
<D2LogicalModel:tppegJunctionExtension> D2LogicalModel:_ExtensionType </D2LogicalModel:tppegJunctionExtension>
[0..1]
</...>

```

#### Schema Component Representation

```

<xs:complexType name="TpegJunction">
  <xs:complexContent>
    <xs:extension base="D2LogicalModel:TpegPoint">
      <xs:sequence>
        <xs:element name="pointCoordinates" type="D2LogicalModel:PointCoordinates"/>
        <xs:element name="name" type="D2LogicalModel:TpegJunctionPointDescriptor" minOccurs="0"/>
        <xs:element name="ilc" type="D2LogicalModel:TpegIlcPointDescriptor" maxOccurs="3"/>
        <xs:element name="otherName" type="D2LogicalModel:TpegOtherPointDescriptor" minOccurs="0"
          maxOccurs="unbounded"/>
        <xs:element name="tppegJunctionExtension" type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

```

[top](#)

### Complex Type: TpegJunctionPointDescriptor

Super-types: [TpegDescriptor](#) < [TpegPointDescriptor](#) (by extension) < **TpegJunctionPointDescriptor** (by extension)

Sub-types: None

**Name** TpegJunctionPointDescriptor  
**Abstract** no  
**Documentation** A descriptor for describing a point at a junction on a road network.

#### XML Instance Representation

```

<...>
  <D2LogicalModel:descriptor> D2LogicalModel:MultilingualString </D2LogicalModel:descriptor> [1] ?
  <D2LogicalModel:tppegDescriptorExtension> D2LogicalModel:_ExtensionType </D2LogicalModel:tppegDescriptorExtension>
  [0..1]
  <D2LogicalModel:tppegPointDescriptorExtension> D2LogicalModel:_ExtensionType
  </D2LogicalModel:tppegPointDescriptorExtension> [0..1]
  <D2LogicalModel:tppegJunctionPointDescriptorType> D2LogicalModel:TpegLoc03JunctionPointDescriptorSubTypeEnum
  </D2LogicalModel:tppegJunctionPointDescriptorType> [1] ?
  <D2LogicalModel:tppegJunctionPointDescriptorExtension> D2LogicalModel:_ExtensionType
  </D2LogicalModel:tppegJunctionPointDescriptorExtension> [0..1]
</...>

```

#### Schema Component Representation

```

<xs:complexType name="TpegJunctionPointDescriptor">
  <xs:complexContent>
    <xs:extension base="D2LogicalModel:TpegPointDescriptor">
      <xs:sequence>
        <xs:element name="tppegJunctionPointDescriptorType"
          type="D2LogicalModel:TpegLoc03JunctionPointDescriptorSubTypeEnum" minOccurs="1" maxOccurs="1"/>
        <xs:element name="tppegJunctionPointDescriptorExtension" type="D2LogicalModel:_ExtensionType"
          minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

```

[top](#)

### Complex Type: TpegLinearLocation

Super-types: None

Sub-types: None

**Name** TpegLinearLocation  
**Abstract** no  
**Documentation** A linear section along a single road defined between two points on the same road by a TPEG-Loc structure.

#### XML Instance Representation

```

<...>
  <D2LogicalModel:tppegDirection> D2LogicalModel:DirectionEnum </D2LogicalModel:tppegDirection> [1] ?
  <D2LogicalModel:tppegLinearLocationType> D2LogicalModel:TpegLoc01LinearLocationSubTypeEnum
  </D2LogicalModel:tppegLinearLocationType> [1] ?
  <D2LogicalModel:to> D2LogicalModel:TpegPoint </D2LogicalModel:to> [1] ?
  <D2LogicalModel:from> D2LogicalModel:TpegPoint </D2LogicalModel:from> [1] ?
  <D2LogicalModel:tppegLinearLocationExtension> D2LogicalModel:_ExtensionType
  </D2LogicalModel:tppegLinearLocationExtension> [0..1]
</...>

```

#### Schema Component Representation

```

<xs:complexType name="TpegLinearLocation">
  <xs:sequence>
    <xs:element name="tppegDirection" type="D2LogicalModel:DirectionEnum" minOccurs="1" maxOccurs="1"/>
    <xs:element name="tppegLinearLocationType" type="D2LogicalModel:TpegLoc01LinearLocationSubTypeEnum"
      minOccurs="1" maxOccurs="1"/>
    <xs:element name="to" type="D2LogicalModel:TpegPoint"/>
    <xs:element name="from" type="D2LogicalModel:TpegPoint"/>
    <xs:element name="tppegLinearLocationExtension" type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>

```

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

[top](#)

## Complex Type: TpegNonJunctionPoint

Super-types: [TpegPoint](#) < TpegNonJunctionPoint (by extension)

Sub-types: None

**Name** TpegNonJunctionPoint  
**Abstract** no  
**Documentation** A point on the road network which is not a road junction point.

### XML Instance Representation

```
<...>
  <D2LogicalModel:tpegPointExtension> D2LogicalModel:_ExtensionType </D2LogicalModel:tpegPointExtension> [0..1]
  <D2LogicalModel:pointCoordinates> D2LogicalModel:PointCoordinates </D2LogicalModel:pointCoordinates> [1]
  <D2LogicalModel:name> D2LogicalModel:TpegOtherPointDescriptor </D2LogicalModel:name> [1..*] ?
  <D2LogicalModel:tpegNonJunctionPointExtension> D2LogicalModel:_ExtensionType
</D2LogicalModel:tpegNonJunctionPointExtension> [0..1]
</...>
```

### Schema Component Representation

```
<xs:complexType name="TpegNonJunctionPoint">
  <xs:complexContent>
    <xs:extension base="D2LogicalModel:TpegPoint">
      <xs:sequence>
        <xs:element name="pointCoordinates" type="D2LogicalModel:PointCoordinates"/>
        <xs:element name="name" type="D2LogicalModel:TpegOtherPointDescriptor" maxOccurs="unbounded"/>
        <xs:element name="tpegNonJunctionPointExtension" type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

[top](#)

## Complex Type: TpegOtherPointDescriptor

Super-types: [TpegDescriptor](#) < [TpegPointDescriptor](#) (by extension) < TpegOtherPointDescriptor (by extension)

Sub-types: None

**Name** TpegOtherPointDescriptor  
**Abstract** no  
**Documentation** General descriptor for describing a point.

### XML Instance Representation

```
<...>
  <D2LogicalModel:descriptor> D2LogicalModel:MultilingualString </D2LogicalModel:descriptor> [1] ?
  <D2LogicalModel:tpegDescriptorExtension> D2LogicalModel:_ExtensionType </D2LogicalModel:tpegDescriptorExtension> [0..1]
  <D2LogicalModel:tpegPointDescriptorExtension> D2LogicalModel:_ExtensionType
</D2LogicalModel:tpegPointDescriptorExtension> [0..1]
  <D2LogicalModel:tpegOtherPointDescriptorType> D2LogicalModel:TpegLoc03OtherPointDescriptorSubtypeEnum
</D2LogicalModel:tpegOtherPointDescriptorType> [1] ?
  <D2LogicalModel:tpegOtherPointDescriptorExtension> D2LogicalModel:_ExtensionType
</D2LogicalModel:tpegOtherPointDescriptorExtension> [0..1]
</...>
```

### Schema Component Representation

```
<xs:complexType name="TpegOtherPointDescriptor">
  <xs:complexContent>
    <xs:extension base="D2LogicalModel:TpegPointDescriptor">
      <xs:sequence>
        <xs:element name="tpegOtherPointDescriptorType"
          type="D2LogicalModel:TpegLoc03OtherPointDescriptorSubtypeEnum" minOccurs="1" maxOccurs="1"/>
        <xs:element name="tpegOtherPointDescriptorExtension" type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

[top](#)

## Complex Type: TpegPoint

Super-types: None

Sub-types:

- [TpegJunction](#) (by extension)
- [TpegNonJunctionPoint](#) (by extension)

**Name** TpegPoint  
**Abstract** yes  
**Documentation** A point on the road network which is either a junction point or a non junction point.

### XML Instance Representation

```
<...>
  <D2LogicalModel:tpegPointExtension> D2LogicalModel:_ExtensionType </D2LogicalModel:tpegPointExtension> [0..1]
</...>
```

#### Schema Component Representation

```
<xs:complexType name="TpegPoint" abstract="true">
  <xs:sequence>
    <xs:element name="tpegPointExtension" type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
```

[top](#)

### Complex Type: TpegPointDescriptor

Super-types: [TpegDescriptor](#) < TpegPointDescriptor (by extension)

Sub-types:

- [TpegJlcPointDescriptor](#) (by extension)
- [TpegJunctionPointDescriptor](#) (by extension)
- [TpegOtherPointDescriptor](#) (by extension)

**Name** TpegPointDescriptor  
**Abstract** yes  
**Documentation** A descriptor for describing a point location.

#### XML Instance Representation

```
<...>
  <D2LogicalModel:descriptor> D2LogicalModel:MultilingualString </D2LogicalModel:descriptor> [1] ?
  <D2LogicalModel:tpegDescriptorExtension> D2LogicalModel:_ExtensionType </D2LogicalModel:tpegDescriptorExtension>
  [0..1]
  <D2LogicalModel:tpegPointDescriptorExtension> D2LogicalModel:_ExtensionType
  </D2LogicalModel:tpegPointDescriptorExtension> [0..1]
</...>
```

#### Schema Component Representation

```
<xs:complexType name="TpegPointDescriptor" abstract="true">
  <xs:complexContent>
    <xs:extension base="D2LogicalModel:TpegDescriptor">
      <xs:sequence>
        <xs:element name="tpegPointDescriptorExtension" type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

[top](#)

### Complex Type: TpegPointLocation

Super-types: None

Sub-types:

- [TpegFramedPoint](#) (by extension)
- [TpegSimplePoint](#) (by extension)

**Name** TpegPointLocation  
**Abstract** yes  
**Documentation** A single point on the road network defined by a TPEG-Loc structure and which has an associated direction of traffic flow.

#### XML Instance Representation

```
<...>
  <D2LogicalModel:tpegDirection> D2LogicalModel:DirectionEnum </D2LogicalModel:tpegDirection> [1] ?
  <D2LogicalModel:tpegPointLocationExtension> D2LogicalModel:_ExtensionType
  </D2LogicalModel:tpegPointLocationExtension> [0..1]
</...>
```

#### Schema Component Representation

```
<xs:complexType name="TpegPointLocation" abstract="true">
  <xs:sequence>
    <xs:element name="tpegDirection" type="D2LogicalModel:DirectionEnum" minOccurs="1" maxOccurs="1"/>
    <xs:element name="tpegPointLocationExtension" type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
```

[top](#)

### Complex Type: TpegSimplePoint

Super-types: [TpegPointLocation](#) < TpegSimplePoint (by extension)

Sub-types: None

**Name** TpegSimplePoint  
**Abstract** no  
**Documentation** A point on the road network which is not bounded by any other points on the road network.

#### XML Instance Representation

```

<...>
<D2LogicalModel:tpegDirection> D2LogicalModel:DirectionEnum </D2LogicalModel:tpegDirection> [1] ?
<D2LogicalModel:tpegPointLocationExtension> D2LogicalModel:_ExtensionType
</D2LogicalModel:tpegPointLocationExtension> [0..1]
<D2LogicalModel:tpegSimplePointLocationType> D2LogicalModel:TpegLoc01SimplePointLocationSubtypeEnum
</D2LogicalModel:tpegSimplePointLocationType> [1] ?
<D2LogicalModel:point> D2LogicalModel:TpegPoint </D2LogicalModel:point> [1] ?
<D2LogicalModel:tpegSimplePointExtension> D2LogicalModel:_ExtensionType </D2LogicalModel:tpegSimplePointExtension>
[0..1]
</...>

```

#### Schema Component Representation

```

<xs:complexType name="TpegSimplePoint">
  <xs:complexContent>
    <xs:extension base="D2LogicalModel:TpegPointLocation">
      <xs:sequence>
        <xs:element name="tpegSimplePointLocationType"
          type="D2LogicalModel:TpegLoc01SimplePointLocationSubtypeEnum" minOccurs="1" maxOccurs="1"/>
        <xs:element name="point" type="D2LogicalModel:TpegPoint"/>
        <xs:element name="tpegSimplePointExtension" type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

```

[top](#)

### Complex Type: TrafficStatusValue

Super-types: [DataValue](#) < [TrafficStatusValue](#) (by extension)  
 Sub-types: None

**Name** TrafficStatusValue  
**Abstract** no  
**Documentation** A measured or calculated value of the status of traffic conditions on a section of road in a specified direction.

#### XML Instance Representation

```

<...
accuracy="D2LogicalModel:Percentage [0..1] ?"
computationalMethod="D2LogicalModel:ComputationMethodEnum [0..1] ?"
numberOfIncompleteInputs="D2LogicalModel:NonNegativeInteger [0..1] ?"
numberOfInputValuesUsed="D2LogicalModel:NonNegativeInteger [0..1] ?"
smoothingFactor="D2LogicalModel:Float [0..1] ?"
standardDeviation="D2LogicalModel:Float [0..1] ?"
supplierCalculatedDataQuality="D2LogicalModel:Percentage [0..1] ?">
  <D2LogicalModel:dataError> D2LogicalModel:Boolean </D2LogicalModel:dataError> [0..1] ?
  <D2LogicalModel:reasonForDataError> D2LogicalModel:MultilingualString </D2LogicalModel:reasonForDataError> [0..1]
  ?
  <D2LogicalModel:dataValueExtension> D2LogicalModel:_ExtensionType </D2LogicalModel:dataValueExtension> [0..1]
  <D2LogicalModel:trafficStatusValue> D2LogicalModel:TrafficStatusEnum </D2LogicalModel:trafficStatusValue> [1] ?
  <D2LogicalModel:trafficStatusValueExtension> D2LogicalModel:_ExtensionType
  </D2LogicalModel:trafficStatusValueExtension> [0..1]
</...>

```

#### Schema Component Representation

```

<xs:complexType name="TrafficStatusValue">
  <xs:complexContent>
    <xs:extension base="D2LogicalModel:DataValue">
      <xs:sequence>
        <xs:element name="trafficStatusValue" type="D2LogicalModel:TrafficStatusEnum" minOccurs="1" maxOccurs="1"/>
        <xs:element name="trafficStatusValueExtension" type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

```

[top](#)

### Complex Type: VehicleCountValue

Super-types: [DataValue](#) < [VehicleCountValue](#) (by extension)  
 Sub-types: None

**Name** VehicleCountValue  
**Abstract** no  
**Documentation** A measured or calculated value of absolute count of vehicles within a specified period of time expressed as non negative integer.

#### XML Instance Representation

```

<...
accuracy="D2LogicalModel:Percentage [0..1] ?"
computationalMethod="D2LogicalModel:ComputationMethodEnum [0..1] ?"
numberOfIncompleteInputs="D2LogicalModel:NonNegativeInteger [0..1] ?"
numberOfInputValuesUsed="D2LogicalModel:NonNegativeInteger [0..1] ?"
smoothingFactor="D2LogicalModel:Float [0..1] ?"
standardDeviation="D2LogicalModel:Float [0..1] ?"
supplierCalculatedDataQuality="D2LogicalModel:Percentage [0..1] ?">
  <D2LogicalModel:dataError> D2LogicalModel:Boolean </D2LogicalModel:dataError> [0..1] ?
  <D2LogicalModel:reasonForDataError> D2LogicalModel:MultilingualString </D2LogicalModel:reasonForDataError> [0..1]
  ?

```

```

</D2LogicalModel:dataValueExtension> D2LogicalModel:_ExtensionType </D2LogicalModel:dataValueExtension> [0..1]
<D2LogicalModel:vehicleCount> D2LogicalModel:NonNegativeInteger </D2LogicalModel:vehicleCount> [1] ?
<D2LogicalModel:vehicleCountValueExtension> D2LogicalModel:_ExtensionType
</D2LogicalModel:vehicleCountValueExtension> [0..1]
</...>

```

#### Schema Component Representation

```

<xs:complexType name="VehicleCountValue">
  <xs:complexContent>
    <xs:extension base="D2LogicalModel:DataValue">
      <xs:sequence>
        <xs:element name="vehicleCount" type="D2LogicalModel:NonNegativeInteger" minOccurs="1" maxOccurs="1"/>
        <xs:element name="vehicleCountValueExtension" type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

```

[top](#)

### Complex Type: VehicleFlowValue

Super-types:	<a href="#">DataValue</a> < <a href="#">VehicleFlowValue</a> (by extension)
Sub-types:	None

<b>Name</b>	VehicleFlowValue
<b>Abstract</b>	no
<b>Documentation</b>	A measured or calculated value of the flow rate of vehicles.

#### XML Instance Representation

```

<...
accuracy="D2LogicalModel:Percentage [0..1] ?"
computationalMethod="D2LogicalModel:ComputationMethodEnum [0..1] ?"
numberOfIncompleteInputs="D2LogicalModel:NonNegativeInteger [0..1] ?"
numberOfInputValuesUsed="D2LogicalModel:NonNegativeInteger [0..1] ?"
smoothingFactor="D2LogicalModel:Float [0..1] ?"
standardDeviation="D2LogicalModel:Float [0..1] ?"
supplierCalculatedDataQuality="D2LogicalModel:Percentage [0..1] ?">
  <D2LogicalModel:dataError> D2LogicalModel:Boolean </D2LogicalModel:dataError> [0..1] ?
  <D2LogicalModel:reasonForDataError> D2LogicalModel:MultilingualString </D2LogicalModel:reasonForDataError> [0..1] ?
  <D2LogicalModel:dataValueExtension> D2LogicalModel:_ExtensionType </D2LogicalModel:dataValueExtension> [0..1]
  <D2LogicalModel:vehicleFlowRate> D2LogicalModel:VehiclesPerHour </D2LogicalModel:vehicleFlowRate> [1] ?
  <D2LogicalModel:vehicleFlowValueExtension> D2LogicalModel:_ExtensionType
  </D2LogicalModel:vehicleFlowValueExtension> [0..1]
</...>

```

#### Schema Component Representation

```

<xs:complexType name="VehicleFlowValue">
  <xs:complexContent>
    <xs:extension base="D2LogicalModel:DataValue">
      <xs:sequence>
        <xs:element name="vehicleFlowRate" type="D2LogicalModel:VehiclesPerHour" minOccurs="1" maxOccurs="1"/>
        <xs:element name="vehicleFlowValueExtension" type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

```

[top](#)

### Complex Type: VersionedReference

Super-types:	None
Sub-types:	<ul style="list-style-type: none"> <li><a href="#">VmsUnitRecordVersionedReference</a> (by extension)</li> <li><a href="#">VmsUnitTableVersionedReference</a> (by extension)</li> </ul>

<b>Name</b>	VersionedReference
<b>Abstract</b>	no

#### XML Instance Representation

```

<...
id="xs:string [1]"
version="xs:string [1]">

```

#### Schema Component Representation

```

<xs:complexType name="VersionedReference">
  <xs:attribute name="id" type="xs:string" use="required"/>
  <xs:attribute name="version" type="xs:string" use="required"/>
</xs:complexType>

```

[top](#)

### Complex Type: Vms

Super-types:	None
Sub-types:	None

<b>Name</b>	Vms
<b>Abstract</b>	no
<b>Documentation</b>	Provides the current status and settings of the VMS and the currently displayed information. Where a VMS is displaying a sequence or alternating set of messages these are ordered according to the messageIndex qualifier.

#### XML Instance Representation

```
<...>
<D2LogicalModel:vmsWorking> D2LogicalModel:Boolean </D2LogicalModel:vmsWorking> [1] ?
<D2LogicalModel:vmsMessageSequencingInterval> D2LogicalModel:Seconds
</D2LogicalModel:vmsMessageSequencingInterval> [0..1] ?
<D2LogicalModel:vmsMessage> D2LogicalModel:_VmsMessageIndexVmsMessage </D2LogicalModel:vmsMessage> [0..*]
<D2LogicalModel:textDisplayAreaSettings> D2LogicalModel:TextDisplayAreaSettings
</D2LogicalModel:textDisplayAreaSettings> [0..1]
<D2LogicalModel:pictogramDisplayAreaSettings>
D2LogicalModel:_VmsPictogramDisplayAreaIndexPictogramDisplayAreaSettings
</D2LogicalModel:pictogramDisplayAreaSettings> [0..*]
<D2LogicalModel:vmsLocationOverride> D2LogicalModel:Location </D2LogicalModel:vmsLocationOverride> [0..1] ?
<D2LogicalModel:managedLogicalLocationOverride> D2LogicalModel:VmsManagedLogicalLocation
</D2LogicalModel:managedLogicalLocationOverride> [0..1] ?
<D2LogicalModel:vmsFault> D2LogicalModel:VmsFault </D2LogicalModel:vmsFault> [0..*]
<D2LogicalModel:vmsExtension> D2LogicalModel:_ExtensionType </D2LogicalModel:vmsExtension> [0..1]
</...>
```

#### Schema Component Representation

```
<xs:complexType name="Vms">
  <xs:sequence>
    <xs:element name="vmsWorking" type="D2LogicalModel:Boolean" minOccurs="1" maxOccurs="1"/>
    <xs:element name="vmsMessageSequencingInterval" type="D2LogicalModel:Seconds" minOccurs="0" maxOccurs="1"/>
    <xs:element name="vmsMessage" type="D2LogicalModel:_VmsMessageIndexVmsMessage" minOccurs="0"
maxOccurs="unbounded"/>
    <xs:element name="textDisplayAreaSettings" type="D2LogicalModel:TextDisplayAreaSettings" minOccurs="0"/>
    <xs:element name="pictogramDisplayAreaSettings"
type="D2LogicalModel:_VmsPictogramDisplayAreaIndexPictogramDisplayAreaSettings" minOccurs="0"
maxOccurs="unbounded"/>
    <xs:element name="vmsLocationOverride" type="D2LogicalModel:Location" minOccurs="0"/>
    <xs:element name="managedLogicalLocationOverride" type="D2LogicalModel:VmsManagedLogicalLocation"
minOccurs="0"/>
    <xs:element name="vmsFault" type="D2LogicalModel:VmsFault" minOccurs="0" maxOccurs="unbounded"/>
    <xs:element name="vmsExtension" type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
```

[top](#)

### Complex Type: VmsFault

<b>Super-types:</b>	<a href="#">Fault</a> < VmsFault (by extension)
<b>Sub-types:</b>	None

<b>Name</b>	VmsFault
<b>Abstract</b>	no
<b>Documentation</b>	Details of the fault which is being reported for the specified variable message sign panel.

#### XML Instance Representation

```
<...>
<D2LogicalModel:faultIdentifier> D2LogicalModel:String </D2LogicalModel:faultIdentifier> [0..1] ?
<D2LogicalModel:faultDescription> D2LogicalModel:String </D2LogicalModel:faultDescription> [0..1] ?
<D2LogicalModel:faultCreationTime> D2LogicalModel:DateTime </D2LogicalModel:faultCreationTime> [0..1] ?
<D2LogicalModel:faultLastUpdateTime> D2LogicalModel:DateTime </D2LogicalModel:faultLastUpdateTime> [1] ?
<D2LogicalModel:faultSeverity> D2LogicalModel:FaultSeverityEnum </D2LogicalModel:faultSeverity> [0..1] ?
<D2LogicalModel:faultExtension> D2LogicalModel:_ExtensionType </D2LogicalModel:faultExtension> [0..1]
<D2LogicalModel:vmsFault> D2LogicalModel:VmsFaultEnum </D2LogicalModel:vmsFault> [1] ?
<D2LogicalModel:vmsFaultExtension> D2LogicalModel:_ExtensionType </D2LogicalModel:vmsFaultExtension> [0..1]
</...>
```

#### Schema Component Representation

```
<xs:complexType name="VmsFault">
  <xs:complexContent>
    <xs:extension base="D2LogicalModel:Fault">
      <xs:sequence>
        <xs:element name="vmsFault" type="D2LogicalModel:VmsFaultEnum" minOccurs="1" maxOccurs="1"/>
        <xs:element name="vmsFaultExtension" type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

[top](#)

### Complex Type: VmsManagedLogicalLocation

<b>Super-types:</b>	None
<b>Sub-types:</b>	None

<b>Name</b>	VmsManagedLogicalLocation
<b>Abstract</b>	no
<b>Documentation</b>	The logical location (e.g. a car park, a section of road, a junction etc.) which a VMS contributes to the management of.



## XML Instance Representation

```
<...>
  <D2LogicalModel:managedLogicalLocation> D2LogicalModel:MultilingualString </D2LogicalModel:managedLogicalLocation>
  [0..1] ?
  <D2LogicalModel:distanceFromLogicalLocation> D2LogicalModel:MetresAsNonNegativeInteger
  </D2LogicalModel:distanceFromLogicalLocation> [0..1] ?
  <D2LogicalModel:managedLocation> D2LogicalModel:Location </D2LogicalModel:managedLocation> [0..1] ?
  <D2LogicalModel:vmsManagedLogicalLocationExtension> D2LogicalModel:_ExtensionType
  </D2LogicalModel:vmsManagedLogicalLocationExtension> [0..1]
</...>
```

## Schema Component Representation

```
<xs:complexType name="VmsManagedLogicalLocation">
  <xs:sequence>
    <xs:element name="managedLogicalLocation" type="D2LogicalModel:MultilingualString" minOccurs="0" maxOccurs="1"/>
    <xs:element name="distanceFromLogicalLocation" type="D2LogicalModel:MetresAsNonNegativeInteger" minOccurs="0"
    maxOccurs="1"/>
    <xs:element name="managedLocation" type="D2LogicalModel:Location" minOccurs="0"/>
    <xs:element name="vmsManagedLogicalLocationExtension" type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
```

[top](#)

## Complex Type: VmsMessage

Super-types:	None
Sub-types:	None

**Name** VmsMessage

**Abstract** no

**Documentation** A message displayed on a VMS which may comprise one or more sequentially displayed text pages and/or pictograms with supplementary details. When in a sequence of displayed messages sequencing of text pages and pictograms within a message are prohibited.

## XML Instance Representation

```
<...>
  <D2LogicalModel:associatedManagementOrDiversionPlan> D2LogicalModel:String
  </D2LogicalModel:associatedManagementOrDiversionPlan> [0..1] ?
  <D2LogicalModel:messageSetBy> D2LogicalModel:MultilingualString </D2LogicalModel:messageSetBy> [0..1] ?
  <D2LogicalModel:setBySystem> D2LogicalModel:Boolean </D2LogicalModel:setBySystem> [0..1] ?
  <D2LogicalModel:reasonForSetting> D2LogicalModel:MultilingualString </D2LogicalModel:reasonForSetting> [0..1] ?
  <D2LogicalModel:codedReasonForSetting> D2LogicalModel:CodedReasonForSettingMessageEnum
  </D2LogicalModel:codedReasonForSetting> [0..1] ?
  <D2LogicalModel:vmsMessageInformationType> D2LogicalModel:VmsMessageInformationTypeEnum
  </D2LogicalModel:vmsMessageInformationType> [0..*] ?
  <D2LogicalModel:primarySetting> D2LogicalModel:Boolean </D2LogicalModel:primarySetting> [0..1] ?
  <D2LogicalModel:mareNostrumCompliant> D2LogicalModel:Boolean </D2LogicalModel:mareNostrumCompliant> [0..1] ?
  <D2LogicalModel:timeLastSet> D2LogicalModel:DateTime </D2LogicalModel:timeLastSet> [1] ?
  <D2LogicalModel:requestedBy> D2LogicalModel:MultilingualString </D2LogicalModel:requestedBy> [0..1] ?
  <D2LogicalModel:situationToWhichMessageIsRelated> D2LogicalModel:VersionedReference
  </D2LogicalModel:situationToWhichMessageIsRelated> [0..1] ?
  <D2LogicalModel:situationRecordToWhichMessageIsRelated> D2LogicalModel:VersionedReference
  </D2LogicalModel:situationRecordToWhichMessageIsRelated> [0..1] ?
  <D2LogicalModel:distanceFromSituationRecord> D2LogicalModel:MetresAsFloat
  </D2LogicalModel:distanceFromSituationRecord> [0..1] ?
  <D2LogicalModel:textPictogramSequencingInterval> D2LogicalModel:Seconds
  </D2LogicalModel:textPictogramSequencingInterval> [0..1] ?
  <D2LogicalModel:textPage> D2LogicalModel:_TextPage </D2LogicalModel:textPage> [0..*]
  <D2LogicalModel:vmsPictogramDisplayArea>
  D2LogicalModel:_VmsMessagePictogramDisplayAreaIndexVmsPictogramDisplayArea
  </D2LogicalModel:vmsPictogramDisplayArea> [0..*]
  <D2LogicalModel:vmsMessageExtension> D2LogicalModel:_ExtensionType </D2LogicalModel:vmsMessageExtension> [0..1]
</...>
```

## Schema Component Representation

```
<xs:complexType name="VmsMessage">
  <xs:sequence>
    <xs:element name="associatedManagementOrDiversionPlan" type="D2LogicalModel:String" minOccurs="0"
    maxOccurs="1"/>
    <xs:element name="messageSetBy" type="D2LogicalModel:MultilingualString" minOccurs="0" maxOccurs="1"/>
    <xs:element name="setBySystem" type="D2LogicalModel:Boolean" minOccurs="0" maxOccurs="1"/>
    <xs:element name="reasonForSetting" type="D2LogicalModel:MultilingualString" minOccurs="0" maxOccurs="1"/>
    <xs:element name="codedReasonForSetting" type="D2LogicalModel:CodedReasonForSettingMessageEnum" minOccurs="0"
    maxOccurs="1"/>
    <xs:element name="vmsMessageInformationType" type="D2LogicalModel:VmsMessageInformationTypeEnum" minOccurs="0"
    maxOccurs="unbounded"/>
    <xs:element name="primarySetting" type="D2LogicalModel:Boolean" minOccurs="0" maxOccurs="1"/>
    <xs:element name="mareNostrumCompliant" type="D2LogicalModel:Boolean" minOccurs="0" maxOccurs="1"/>
    <xs:element name="timeLastSet" type="D2LogicalModel:DateTime" minOccurs="1" maxOccurs="1"/>
    <xs:element name="requestedBy" type="D2LogicalModel:MultilingualString" minOccurs="0" maxOccurs="1"/>
    <xs:element name="situationToWhichMessageIsRelated" type="D2LogicalModel:VersionedReference" minOccurs="0"
    maxOccurs="1"/>
    <xs:element name="situationRecordToWhichMessageIsRelated" type="D2LogicalModel:VersionedReference" minOccurs="0"
    maxOccurs="1"/>
    <xs:element name="distanceFromSituationRecord" type="D2LogicalModel:MetresAsFloat" minOccurs="0" maxOccurs="1"/>
    <xs:element name="textPictogramSequencingInterval" type="D2LogicalModel:Seconds" minOccurs="0" maxOccurs="1"/>
    <xs:element name="textPage" type="D2LogicalModel:_TextPage" minOccurs="0" maxOccurs="unbounded"/>
    <xs:element name="vmsPictogramDisplayArea"
    type="D2LogicalModel:_VmsMessagePictogramDisplayAreaIndexVmsPictogramDisplayArea" minOccurs="0"
    maxOccurs="unbounded"/>
    <xs:element name="vmsMessageExtension" type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
```



**Complex Type: VmsPictogram**

Super-types:	None
Sub-types:	None

<b>Name</b>	VmsPictogram
<b>Abstract</b>	no
<b>Documentation</b>	A main pictogram displayable on the VMS panel. Note a main pictogram may have an associated supplementary panel which may itself contain a further pictogram and line of text.

**XML Instance Representation**

```
<...>
<D2LogicalModel:pictogramDescription> D2LogicalModel:VmsDatexPictogramEnum </D2LogicalModel:pictogramDescription>
[0..*] ?
<D2LogicalModel:pictogramCode> D2LogicalModel:String </D2LogicalModel:pictogramCode> [0..1] ?
<D2LogicalModel:pictogramUrl> D2LogicalModel:Url </D2LogicalModel:pictogramUrl> [0..1] ?
<D2LogicalModel:additionalPictogramDescription> D2LogicalModel:MultilingualString
</D2LogicalModel:additionalPictogramDescription> [0..1] ?
<D2LogicalModel:pictogramFlashing> D2LogicalModel:Boolean </D2LogicalModel:pictogramFlashing> [0..1] ?
<D2LogicalModel:pictogramInverseColour> D2LogicalModel:Boolean </D2LogicalModel:pictogramInverseColour> [0..1]
?
<D2LogicalModel:presenceOfRedTriangle> D2LogicalModel:Boolean </D2LogicalModel:presenceOfRedTriangle> [1] ?
<D2LogicalModel:viennaConventionCompliant> D2LogicalModel:Boolean </D2LogicalModel:viennaConventionCompliant>
[0..1] ?
<D2LogicalModel:distanceAttribute> D2LogicalModel:MetresAsNonNegativeInteger </D2LogicalModel:distanceAttribute>
[0..1] ?
<D2LogicalModel:heightAttribute> D2LogicalModel:MetresAsFloat </D2LogicalModel:heightAttribute> [0..1] ?
<D2LogicalModel:lengthAttribute> D2LogicalModel:MetresAsFloat </D2LogicalModel:lengthAttribute> [0..1] ?
<D2LogicalModel:speedAttribute> D2LogicalModel:KilometresPerHour </D2LogicalModel:speedAttribute> [0..1] ?
<D2LogicalModel:weightAttribute> D2LogicalModel:Tonnes </D2LogicalModel:weightAttribute> [0..1] ?
<D2LogicalModel:weightPerAxleAttribute> D2LogicalModel:Tonnes </D2LogicalModel:weightPerAxleAttribute> [0..1] ?
<D2LogicalModel:widthAttribute> D2LogicalModel:MetresAsFloat </D2LogicalModel:widthAttribute> [0..1] ?
<D2LogicalModel:vmsSupplementaryPanel> D2LogicalModel:VmsSupplementaryPanel
</D2LogicalModel:vmsSupplementaryPanel> [0..1]
<D2LogicalModel:vmsPictogramExtension> D2LogicalModel:_ExtensionType </D2LogicalModel:vmsPictogramExtension>
[0..1]
</...>
```

**Schema Component Representation**

```
<xs:complexType name="VmsPictogram">
  <xs:sequence>
    <xs:element name="pictogramDescription" type="D2LogicalModel:VmsDatexPictogramEnum" minOccurs="0"
maxOccurs="unbounded"/>
    <xs:element name="pictogramCode" type="D2LogicalModel:String" minOccurs="0" maxOccurs="1"/>
    <xs:element name="pictogramUrl" type="D2LogicalModel:Url" minOccurs="0" maxOccurs="1"/>
    <xs:element name="additionalPictogramDescription" type="D2LogicalModel:MultilingualString" minOccurs="0"
maxOccurs="1"/>
    <xs:element name="pictogramFlashing" type="D2LogicalModel:Boolean" minOccurs="0" maxOccurs="1"/>
    <xs:element name="pictogramInverseColour" type="D2LogicalModel:Boolean" minOccurs="0" maxOccurs="1"/>
    <xs:element name="presenceOfRedTriangle" type="D2LogicalModel:Boolean" minOccurs="1" maxOccurs="1"/>
    <xs:element name="viennaConventionCompliant" type="D2LogicalModel:Boolean" minOccurs="0" maxOccurs="1"/>
    <xs:element name="distanceAttribute" type="D2LogicalModel:MetresAsNonNegativeInteger" minOccurs="0"
maxOccurs="1"/>
    <xs:element name="heightAttribute" type="D2LogicalModel:MetresAsFloat" minOccurs="0" maxOccurs="1"/>
    <xs:element name="lengthAttribute" type="D2LogicalModel:MetresAsFloat" minOccurs="0" maxOccurs="1"/>
    <xs:element name="speedAttribute" type="D2LogicalModel:KilometresPerHour" minOccurs="0" maxOccurs="1"/>
    <xs:element name="weightAttribute" type="D2LogicalModel:Tonnes" minOccurs="0" maxOccurs="1"/>
    <xs:element name="weightPerAxleAttribute" type="D2LogicalModel:Tonnes" minOccurs="0" maxOccurs="1"/>
    <xs:element name="widthAttribute" type="D2LogicalModel:MetresAsFloat" minOccurs="0" maxOccurs="1"/>
    <xs:element name="vmsSupplementaryPanel" type="D2LogicalModel:VmsSupplementaryPanel" minOccurs="0"/>
    <xs:element name="vmsPictogramExtension" type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
```

**Complex Type: VmsPictogramDisplayArea**

Super-types:	None
Sub-types:	None

<b>Name</b>	VmsPictogramDisplayArea
<b>Abstract</b>	no
<b>Documentation</b>	An area on a VMS used for the display of pictograms and associated supplemental information or instructions.

**XML Instance Representation**

```
<...>
<D2LogicalModel:synchronizedSequencingWithTextPages> D2LogicalModel:Boolean
</D2LogicalModel:synchronizedSequencingWithTextPages> [0..1] ?
<D2LogicalModel:vmsPictogram> D2LogicalModel:_VmsPictogramDisplayAreaPictogramSequencingIndexVmsPictogram
</D2LogicalModel:vmsPictogram> [0..*]
<D2LogicalModel:vmsPictogramDisplayAreaExtension> D2LogicalModel:_ExtensionType
</D2LogicalModel:vmsPictogramDisplayAreaExtension> [0..1]
</...>
```

**Schema Component Representation**

```
<xs:complexType name="VmsPictogramDisplayArea">
```

```

<xs:sequence>
  <xs:element name="synchronizedSequencingWithTextPages" type="D2LogicalModel:Boolean" minOccurs="0"
maxOccurs="1"/>
  <xs:element name="vmsPictogram"
type="D2LogicalModel:_VmsPictogramDisplayAreaPictogramSequencingIndexVmsPictogram" minOccurs="0"
maxOccurs="unbounded"/>
  <xs:element name="vmsPictogramDisplayAreaExtension" type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
</xs:sequence>
</xs:complexType>

```

[top](#)

## Complex Type: VmsPublication

Super-types: [PayloadPublication](#) < VmsPublication (by extension)

Sub-types: None

<b>Name</b>	VmsPublication
<b>Abstract</b>	no
<b>Documentation</b>	A publication containing the current status and settings of one or more VMS units, each unit controlling one or more individual variable message signs.

### XML Instance Representation

```

<...
  lang="D2LogicalModel:Language [1] ?">
  <D2LogicalModel:publicationTime> D2LogicalModel:DateTime </D2LogicalModel:publicationTime> [1] ?
  <D2LogicalModel:publicationCreator> D2LogicalModel:InternationalIdentifier </D2LogicalModel:publicationCreator>
[1]
  <D2LogicalModel:payloadPublicationExtension> D2LogicalModel:_ExtensionType
</D2LogicalModel:payloadPublicationExtension> [0..1]
  <D2LogicalModel:headerInformation> D2LogicalModel:HeaderInformation </D2LogicalModel:headerInformation> [1]
  <D2LogicalModel:vmsUnit> D2LogicalModel:VmsUnit </D2LogicalModel:vmsUnit> [1..*]
  <D2LogicalModel:vmsPublicationExtension> D2LogicalModel:_ExtensionType </D2LogicalModel:vmsPublicationExtension>
[0..1]
</...>

```

### Schema Component Representation

```

<xs:complexType name="VmsPublication">
  <xs:complexContent>
    <xs:extension base="D2LogicalModel:PayloadPublication">
      <xs:sequence>
        <xs:element name="headerInformation" type="D2LogicalModel:HeaderInformation"/>
        <xs:element name="vmsUnit" type="D2LogicalModel:VmsUnit" maxOccurs="unbounded"/>
        <xs:element name="vmsPublicationExtension" type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

```

[top](#)

## Complex Type: VmsSetting

Super-types: None

Sub-types: 

- [VmsUnit](#) (by extension)

<b>Name</b>	VmsSetting
<b>Abstract</b>	yes
<b>Documentation</b>	Provides information on variable message signs and the information currently displayed.

### XML Instance Representation

```

<...>
  <D2LogicalModel:vmsSettingExtension> D2LogicalModel:_ExtensionType </D2LogicalModel:vmsSettingExtension> [0..1]
</...>

```

### Schema Component Representation

```

<xs:complexType name="VmsSetting" abstract="true">
  <xs:sequence>
    <xs:element name="vmsSettingExtension" type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>

```

[top](#)

## Complex Type: VmsSupplementaryPanel

Super-types: None

Sub-types: None

<b>Name</b>	VmsSupplementaryPanel
<b>Abstract</b>	no
<b>Documentation</b>	A panel which may display information or a regulatory instruction which is supplemental to the associated pictogram, comprising either an additional line of text or a pictogram or both.

### XML Instance Representation

```

<...>
<D2LogicalModel:supplementaryMessageDescription> D2LogicalModel:MultilingualString
</D2LogicalModel:supplementaryMessageDescription> [0..1] ?
<D2LogicalModel:vmsSupplementaryPictogram> D2LogicalModel:VmsSupplementaryPictogram
</D2LogicalModel:vmsSupplementaryPictogram> [0..1]
<D2LogicalModel:vmsSupplementaryText> D2LogicalModel:VmsTextLine </D2LogicalModel:vmsSupplementaryText> [0..1] ?
<D2LogicalModel:vmsSupplementaryPanelExtension> D2LogicalModel:ExtensionType
</D2LogicalModel:vmsSupplementaryPanelExtension> [0..1]
</...>

```

#### Schema Component Representation

```

<xs:complexType name="VmsSupplementaryPanel">
  <xs:sequence>
    <xs:element name="supplementaryMessageDescription" type="D2LogicalModel:MultilingualString" minOccurs="0"
      maxOccurs="1"/>
    <xs:element name="vmsSupplementaryPictogram" type="D2LogicalModel:VmsSupplementaryPictogram" minOccurs="0"/>
    <xs:element name="vmsSupplementaryText" type="D2LogicalModel:VmsTextLine" minOccurs="0"/>
    <xs:element name="vmsSupplementaryPanelExtension" type="D2LogicalModel:ExtensionType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>

```

[top](#)

### Complex Type: VmsSupplementaryPictogram

Super-types:	None
Sub-types:	None

<b>Name</b>	VmsSupplementaryPictogram
<b>Abstract</b>	no
<b>Documentation</b>	An additional pictogram that is displayed in the panel which is supplemental to the associated pictogram display.

#### XML Instance Representation

```

<...>
<D2LogicalModel:supplementaryPictogramDescription> D2LogicalModel:VmsDatexSupplementalPictogramEnum
</D2LogicalModel:supplementaryPictogramDescription> [0..1] ?
<D2LogicalModel:supplementaryPictogramCode> D2LogicalModel:String </D2LogicalModel:supplementaryPictogramCode>
[0..1] ?
<D2LogicalModel:supplementaryPictogramUrl> D2LogicalModel:Url </D2LogicalModel:supplementaryPictogramUrl> [0..1] ?
<D2LogicalModel:additionalSupplementaryPictogramDescription> D2LogicalModel:MultilingualString
</D2LogicalModel:additionalSupplementaryPictogramDescription> [0..1] ?
<D2LogicalModel:pictogramFlashing> D2LogicalModel:Boolean </D2LogicalModel:pictogramFlashing> [0..1] ?
<D2LogicalModel:vmsSupplementaryPictogramExtension> D2LogicalModel:ExtensionType
</D2LogicalModel:vmsSupplementaryPictogramExtension> [0..1]
</...>

```

#### Schema Component Representation

```

<xs:complexType name="VmsSupplementaryPictogram">
  <xs:sequence>
    <xs:element name="supplementaryPictogramDescription" type="D2LogicalModel:VmsDatexSupplementalPictogramEnum"
      minOccurs="0" maxOccurs="1"/>
    <xs:element name="supplementaryPictogramCode" type="D2LogicalModel:String" minOccurs="0" maxOccurs="1"/>
    <xs:element name="supplementaryPictogramUrl" type="D2LogicalModel:Url" minOccurs="0" maxOccurs="1"/>
    <xs:element name="additionalSupplementaryPictogramDescription" type="D2LogicalModel:MultilingualString"
      minOccurs="0" maxOccurs="1"/>
    <xs:element name="pictogramFlashing" type="D2LogicalModel:Boolean" minOccurs="0" maxOccurs="1"/>
    <xs:element name="vmsSupplementaryPictogramExtension" type="D2LogicalModel:ExtensionType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>

```

[top](#)

### Complex Type: VmsText

Super-types:	None
Sub-types:	None

<b>Name</b>	VmsText
<b>Abstract</b>	no
<b>Documentation</b>	A page of text (comprising one or more ordered lines) that are displayed simultaneously on the VMS. Where more than one page is defined these are sequentially displayed according to their "pageNumber".

#### XML Instance Representation

```

<...>
<D2LogicalModel:vmsLegendCode> D2LogicalModel:String </D2LogicalModel:vmsLegendCode> [0..1] ?
<D2LogicalModel:vmsTextImageUrl> D2LogicalModel:Url </D2LogicalModel:vmsTextImageUrl> [0..1] ?
<D2LogicalModel:vmsTextLine> D2LogicalModel:VmsTextLineIndexVmsTextLine </D2LogicalModel:vmsTextLine> [0..*]
<D2LogicalModel:vmsTextExtension> D2LogicalModel:ExtensionType </D2LogicalModel:vmsTextExtension> [0..1]
</...>

```

#### Schema Component Representation

```

<xs:complexType name="VmsText">
  <xs:sequence>
    <xs:element name="vmsLegendCode" type="D2LogicalModel:String" minOccurs="0" maxOccurs="1"/>
    <xs:element name="vmsTextImageUrl" type="D2LogicalModel:Url" minOccurs="0" maxOccurs="1"/>
    <xs:element name="vmsTextLine" type="D2LogicalModel:VmsTextLineIndexVmsTextLine" minOccurs="0"
      maxOccurs="unbounded"/>
    <xs:element name="vmsTextExtension" type="D2LogicalModel:ExtensionType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>

```

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

[top](#)

## Complex Type: VmsTextLine

Super-types: None  
Sub-types: None

Name VmsTextLine  
Abstract no  
Documentation A single line of text on a text display area or supplementary panel.

### XML Instance Representation

```
<...>
<D2LogicalModel:vmsTextLine> D2LogicalModel:String </D2LogicalModel:vmsTextLine> [1] ?
<D2LogicalModel:vmsTextLineLanguage> D2LogicalModel:Language </D2LogicalModel:vmsTextLineLanguage> [0..1] ?
<D2LogicalModel:vmsTextLineColour> D2LogicalModel:ColourEnum </D2LogicalModel:vmsTextLineColour> [0..1] ?
<D2LogicalModel:vmsTextLineFlashing> D2LogicalModel:Boolean </D2LogicalModel:vmsTextLineFlashing> [0..1] ?
<D2LogicalModel:vmsTextLineHtml> D2LogicalModel:String </D2LogicalModel:vmsTextLineHtml> [0..1] ?
<D2LogicalModel:vmsTextLineExtension> D2LogicalModel:_ExtensionType </D2LogicalModel:vmsTextLineExtension> [0..1]
</...>
```

### Schema Component Representation

```
<xs:complexType name="VmsTextLine">
  <xs:sequence>
    <xs:element name="vmsTextLine" type="D2LogicalModel:String" minOccurs="1" maxOccurs="1"/>
    <xs:element name="vmsTextLineLanguage" type="D2LogicalModel:Language" minOccurs="0" maxOccurs="1"/>
    <xs:element name="vmsTextLineColour" type="D2LogicalModel:ColourEnum" minOccurs="0" maxOccurs="1"/>
    <xs:element name="vmsTextLineFlashing" type="D2LogicalModel:Boolean" minOccurs="0" maxOccurs="1"/>
    <xs:element name="vmsTextLineHtml" type="D2LogicalModel:String" minOccurs="0" maxOccurs="1"/>
    <xs:element name="vmsTextLineExtension" type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
```

[top](#)

## Complex Type: VmsUnit

Super-types: [VmsSetting](#) < VmsUnit (by extension)  
Sub-types: None

Name VmsUnit  
Abstract no  
Documentation Status of a VMS unit which may control one or more variable message signs on a single gantry or on different gantries.

### XML Instance Representation

```
<...>
<D2LogicalModel:vmsSettingExtension> D2LogicalModel:_ExtensionType </D2LogicalModel:vmsSettingExtension> [0..1]
<D2LogicalModel:vmsUnitTableReference> D2LogicalModel:_VmsUnitTableVersionedReference
</D2LogicalModel:vmsUnitTableReference> [1] ?
<D2LogicalModel:vmsUnitReference> D2LogicalModel:_VmsUnitRecordVersionedReference
</D2LogicalModel:vmsUnitReference> [1] ?
<D2LogicalModel:vms> D2LogicalModel:_VmsUnitVmsIndexVms </D2LogicalModel:vms> [0..*]
<D2LogicalModel:vmsUnitFault> D2LogicalModel:VmsUnitFault </D2LogicalModel:vmsUnitFault> [0..*]
<D2LogicalModel:vmsUnitExtension> D2LogicalModel:_ExtensionType </D2LogicalModel:vmsUnitExtension> [0..1]
</...>
```

### Schema Component Representation

```
<xs:complexType name="VmsUnit">
  <xs:complexContent>
    <xs:extension base="D2LogicalModel:VmsSetting">
      <xs:sequence>
        <xs:element name="vmsUnitTableReference" type="D2LogicalModel:_VmsUnitTableVersionedReference"
          minOccurs="1" maxOccurs="1"/>
        <xs:element name="vmsUnitReference" type="D2LogicalModel:_VmsUnitRecordVersionedReference" minOccurs="1"
          maxOccurs="1"/>
        <xs:element name="vms" type="D2LogicalModel:_VmsUnitVmsIndexVms" minOccurs="0" maxOccurs="unbounded"/>
        <xs:element name="vmsUnitFault" type="D2LogicalModel:VmsUnitFault" minOccurs="0" maxOccurs="unbounded"/>
        <xs:element name="vmsUnitExtension" type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

[top](#)

## Complex Type: VmsUnitFault

Super-types: [Fault](#) < VmsUnitFault (by extension)  
Sub-types: None

Name VmsUnitFault  
Abstract no  
Documentation Details of the fault which is being reported for the specified variable message sign control unit.

## XML Instance Representation

```
<...>
  <D2LogicalModel:faultIdentifier> D2LogicalModel:String </D2LogicalModel:faultIdentifier> [0..1] ?
  <D2LogicalModel:faultDescription> D2LogicalModel:String </D2LogicalModel:faultDescription> [0..1] ?
  <D2LogicalModel:faultCreationTime> D2LogicalModel:DateTime </D2LogicalModel:faultCreationTime> [0..1] ?
  <D2LogicalModel:faultLastUpdateTime> D2LogicalModel:DateTime </D2LogicalModel:faultLastUpdateTime> [1] ?
  <D2LogicalModel:faultSeverity> D2LogicalModel:FaultSeverityEnum </D2LogicalModel:faultSeverity> [0..1] ?
  <D2LogicalModel:faultExtension> D2LogicalModel:_ExtensionType </D2LogicalModel:faultExtension> [0..1]
  <D2LogicalModel:vmsUnitFault> D2LogicalModel:VmsFaultEnum </D2LogicalModel:vmsUnitFault> [1] ?
  <D2LogicalModel:vmsUnitFaultExtension> D2LogicalModel:_ExtensionType </D2LogicalModel:vmsUnitFaultExtension>
  [0..1]
</...>
```

## Schema Component Representation

```
<xs:complexType name="VmsUnitFault">
  <xs:complexContent>
    <xs:extension base="D2LogicalModel:Fault">
      <xs:sequence>
        <xs:element name="vmsUnitFault" type="D2LogicalModel:VmsFaultEnum" minOccurs="1" maxOccurs="1"/>
        <xs:element name="vmsUnitFaultExtension" type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

[top](#)

## Complex Type: \_ExtensionType

Super-types:	None
Sub-types:	None

Name \_ExtensionType

**Abstract** no

## XML Instance Representation

```
<...>
  Allow any elements from any namespace (lax validation). [0..*]
</...>
```

## Schema Component Representation

```
<xs:complexType name="_ExtensionType">
  <xs:sequence>
    <xs:any namespace="##any" processContents="lax" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>
```

[top](#)

## Complex Type: \_IntermediatePointOnLinearElement

Super-types:	None
Sub-types:	None

Name \_IntermediatePointOnLinearElement

**Abstract** no

## XML Instance Representation

```
<...
  index="xs:int [1]">
  <D2LogicalModel:referent> D2LogicalModel:Referent </D2LogicalModel:referent> [1]
</...>
```

## Schema Component Representation

```
<xs:complexType name="_IntermediatePointOnLinearElement">
  <xs:sequence>
    <xs:element name="referent" type="D2LogicalModel:Referent" minOccurs="1" maxOccurs="1"/>
  </xs:sequence>
  <xs:attribute name="index" type="xs:int" use="required"/>
</xs:complexType>
```

[top](#)

## Complex Type: \_LinearExtensionType

Super-types:	None
Sub-types:	None

Name \_LinearExtensionType

**Abstract** no

## XML Instance Representation

```
<...>
  <D2LogicalModel:openlrExtendedLinear> D2LogicalModel:OpenlrExtendedLinear </D2LogicalModel:openlrExtendedLinear>
  [0..1]
</...>
```

```
Allow any elements from a namespace other than this schema's namespace (lax validation). [0..*]  
</...>
```

#### Schema Component Representation

```
<xs:complexType name="_LinearExtensionType">  
  <xs:sequence>  
    <xs:element name="openlrExtendedLinear" type="D2LogicalModel:OpenlrExtendedLinear" minOccurs="0"/>  
    <xs:any namespace="##other" processContents="lax" minOccurs="0" maxOccurs="unbounded"/>  
  </xs:sequence>  
</xs:complexType>
```

[top](#)

### Complex Type: **\_PointExtensionType**

Super-types:	None
Sub-types:	None

**Name** \_PointExtensionType  
**Abstract** no

#### XML Instance Representation

```
<...>  
<D2LogicalModel:openlrExtendedPoint> D2LogicalModel:OpenlrExtendedPoint </D2LogicalModel:openlrExtendedPoint>  
[0..1]  
<D2LogicalModel:pointExtended> D2LogicalModel:PointExtended </D2LogicalModel:pointExtended> [0..1]  
Allow any elements from a namespace other than this schema's namespace (lax validation). [0..*]  
</...>
```

#### Schema Component Representation

```
<xs:complexType name="_PointExtensionType">  
  <xs:sequence>  
    <xs:element name="openlrExtendedPoint" type="D2LogicalModel:OpenlrExtendedPoint" minOccurs="0"/>  
    <xs:element name="pointExtended" type="D2LogicalModel:PointExtended" minOccurs="0"/>  
    <xs:any namespace="##other" processContents="lax" minOccurs="0" maxOccurs="unbounded"/>  
  </xs:sequence>  
</xs:complexType>
```

[top](#)

### Complex Type: **\_TextPage**

Super-types:	None
Sub-types:	None

**Name** \_TextPage  
**Abstract** no

#### XML Instance Representation

```
<...  
  pageNumber="xs:int [1]">  
<D2LogicalModel:vmsText> D2LogicalModel:VmsText </D2LogicalModel:vmsText> [1]  
</...>
```

#### Schema Component Representation

```
<xs:complexType name="_TextPage">  
  <xs:sequence>  
    <xs:element name="vmsText" type="D2LogicalModel:VmsText" minOccurs="1" maxOccurs="1"/>  
  </xs:sequence>  
  <xs:attribute name="pageNumber" type="xs:int" use="required"/>  
</xs:complexType>
```

[top](#)

### Complex Type: **\_VmsMessageIndexVmsMessage**

Super-types:	None
Sub-types:	None

**Name** \_VmsMessageIndexVmsMessage  
**Abstract** no

#### XML Instance Representation

```
<...  
  messageIndex="xs:int [1]">  
<D2LogicalModel:vmsMessage> D2LogicalModel:VmsMessage </D2LogicalModel:vmsMessage> [1]  
</...>
```

#### Schema Component Representation

```
<xs:complexType name="_VmsMessageIndexVmsMessage">  
  <xs:sequence>  
    <xs:element name="vmsMessage" type="D2LogicalModel:VmsMessage" minOccurs="1" maxOccurs="1"/>  
  </xs:sequence>  
  <xs:attribute name="messageIndex" type="xs:int" use="required"/>  
</xs:complexType>
```

**Complex Type: \_VmsMessagePictogramDisplayAreaIndexVmsPictogramDisplayArea**

Super-types: None  
 Sub-types: None

Name \_VmsMessagePictogramDisplayAreaIndexVmsPictogramDisplayArea  
 Abstract no

**XML Instance Representation**

```
<...
  pictogramDisplayAreaIndex="xs:int [1]">
    <D2LogicalModel:vmsPictogramDisplayArea> D2LogicalModel:VmsPictogramDisplayArea
  </D2LogicalModel:vmsPictogramDisplayArea> [1]
</...>
```

**Schema Component Representation**

```
<xs:complexType name="_VmsMessagePictogramDisplayAreaIndexVmsPictogramDisplayArea">
  <xs:sequence>
    <xs:element name="vmsPictogramDisplayArea" type="D2LogicalModel:VmsPictogramDisplayArea" minOccurs="1"
      maxOccurs="1"/>
  </xs:sequence>
  <xs:attribute name="pictogramDisplayAreaIndex" type="xs:int" use="required"/>
</xs:complexType>
```

**Complex Type: \_VmsPictogramDisplayAreaIndexPictogramDisplayAreaSettings**

Super-types: None  
 Sub-types: None

Name \_VmsPictogramDisplayAreaIndexPictogramDisplayAreaSettings  
 Abstract no

**XML Instance Representation**

```
<...
  pictogramDisplayAreaIndex="xs:int [1]">
    <D2LogicalModel:pictogramDisplayAreaSettings> D2LogicalModel:PictogramDisplayAreaSettings
  </D2LogicalModel:pictogramDisplayAreaSettings> [1]
</...>
```

**Schema Component Representation**

```
<xs:complexType name="_VmsPictogramDisplayAreaIndexPictogramDisplayAreaSettings">
  <xs:sequence>
    <xs:element name="pictogramDisplayAreaSettings" type="D2LogicalModel:PictogramDisplayAreaSettings" minOccurs="1"
      maxOccurs="1"/>
  </xs:sequence>
  <xs:attribute name="pictogramDisplayAreaIndex" type="xs:int" use="required"/>
</xs:complexType>
```

**Complex Type: \_VmsPictogramDisplayAreaPictogramSequencingIndexVmsPictogram**

Super-types: None  
 Sub-types: None

Name \_VmsPictogramDisplayAreaPictogramSequencingIndexVmsPictogram  
 Abstract no

**XML Instance Representation**

```
<...
  pictogramSequencingIndex="xs:int [1]">
    <D2LogicalModel:vmsPictogram> D2LogicalModel:VmsPictogram </D2LogicalModel:vmsPictogram> [1]
</...>
```

**Schema Component Representation**

```
<xs:complexType name="_VmsPictogramDisplayAreaPictogramSequencingIndexVmsPictogram">
  <xs:sequence>
    <xs:element name="vmsPictogram" type="D2LogicalModel:VmsPictogram" minOccurs="1" maxOccurs="1"/>
  </xs:sequence>
  <xs:attribute name="pictogramSequencingIndex" type="xs:int" use="required"/>
</xs:complexType>
```

**Complex Type: \_VmsTextLineIndexVmsTextLine**

Super-types: None  
 Sub-types: None

Name \_VmsTextLineIndexVmsTextLine



**Abstract**

no

**XML Instance Representation**

```
<...  
  lineIndex="xs:int [1]">  
  <D2LogicalModel:vmsTextLine> D2LogicalModel:VmsTextLine </D2LogicalModel:vmsTextLine> [1]  
</...>
```

**Schema Component Representation**

```
<xs:complexType name="_VmsTextLineIndexVmsTextLine">  
  <xs:sequence>  
    <xs:element name="vmsTextLine" type="D2LogicalModel:VmsTextLine" minOccurs="1" maxOccurs="1"/>  
  </xs:sequence>  
  <xs:attribute name="lineIndex" type="xs:int" use="required"/>  
</xs:complexType>
```

[top](#)

**Complex Type: VmsUnitRecordVersionedReference**

Super-types:	<a href="#">VersionedReference</a> < <u>VmsUnitRecordVersionedReference</u> (by extension)
Sub-types:	None

**Name** VmsUnitRecordVersionedReference

**Abstract** no

**XML Instance Representation**

```
<...  
  id="xs:string [1]"  
  version="xs:string [1]"  
  targetClass="VmsUnitRecord [1]" />
```

**Schema Component Representation**

```
<xs:complexType name="_VmsUnitRecordVersionedReference">  
  <xs:complexContent>  
    <xs:extension base="D2LogicalModel:VersionedReference">  
      <xs:attribute name="targetClass" use="required" fixed="VmsUnitRecord"/>  
    </xs:extension>  
  </xs:complexContent>  
</xs:complexType>
```

[top](#)

**Complex Type: VmsUnitTableVersionedReference**

Super-types:	<a href="#">VersionedReference</a> < <u>VmsUnitTableVersionedReference</u> (by extension)
Sub-types:	None

**Name** VmsUnitTableVersionedReference

**Abstract** no

**XML Instance Representation**

```
<...  
  id="xs:string [1]"  
  version="xs:string [1]"  
  targetClass="VmsUnitTable [1]" />
```

**Schema Component Representation**

```
<xs:complexType name="_VmsUnitTableVersionedReference">  
  <xs:complexContent>  
    <xs:extension base="D2LogicalModel:VersionedReference">  
      <xs:attribute name="targetClass" use="required" fixed="VmsUnitTable"/>  
    </xs:extension>  
  </xs:complexContent>  
</xs:complexType>
```

[top](#)

**Complex Type: VmsUnitVmsIndexVms**

Super-types:	None
Sub-types:	None

**Name** VmsUnitVmsIndexVms

**Abstract** no

**XML Instance Representation**

```
<...  
  vmsIndex="xs:int [1]">  
  <D2LogicalModel:vms> D2LogicalModel:Vms </D2LogicalModel:vms> [1]  
</...>
```

**Schema Component Representation**

```
<xs:complexType name="_VmsUnitVmsIndexVms">  
  <xs:sequence>
```

```
<xs:element name="vms" type="D2LogicalModel:Vms" minOccurs="1" maxOccurs="1"/>
</xs:sequence>
<xs:attribute name="vmsIndex" type="xs:int" use="required"/>
</xs:complexType>
```

[top](#)

## Simple Type: AlertCDirectionEnum

Super-types: [xs:string](#) < [AlertCDirectionEnum](#) (by restriction)

Sub-types: None

Name AlertCDirectionEnum

Content

- Base XSD Type: string
- value comes from list: {'both'|'negative'|'positive'|'unknown'}

Documentation

The direction of traffic flow concerned by a situation or traffic data. In ALERT-C the positive (resp. negative) direction corresponds to the positive offset direction within the RDS location table.

### Schema Component Representation

```
<xs:simpleType name="AlertCDirectionEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="both"/>
    <xs:enumeration value="negative"/>
    <xs:enumeration value="positive"/>
    <xs:enumeration value="unknown"/>
  </xs:restriction>
</xs:simpleType>
```

[top](#)

## Simple Type: AlertCLocationCode

Super-types: [xs:nonNegativeInteger](#) < [NonNegativeInteger](#) (by restriction) < [AlertCLocationCode](#) (by restriction)

Sub-types: None

Name AlertCLocationCode

Content

- Base XSD Type: nonNegativeInteger

Documentation

A positive integer number (between 1 and 63,487) which uniquely identifies a pre-defined Alert C location defined within an Alert-C table.

### Schema Component Representation

```
<xs:simpleType name="AlertCLocationCode">
  <xs:restriction base="D2LogicalModel:NonNegativeInteger"/>
</xs:simpleType>
```

[top](#)

## Simple Type: AngleInDegrees

Super-types: [xs:nonNegativeInteger](#) < [NonNegativeInteger](#) (by restriction) < [AngleInDegrees](#) (by restriction)

Sub-types: None

Name AngleInDegrees

Content

- Base XSD Type: nonNegativeInteger

Documentation

An integer number representing an angle in whole degrees between 0 and 359.

### Schema Component Representation

```
<xs:simpleType name="AngleInDegrees">
  <xs:restriction base="D2LogicalModel:NonNegativeInteger"/>
</xs:simpleType>
```

[top](#)

## Simple Type: AreaOfInterestEnum

Super-types: [xs:string](#) < [AreaOfInterestEnum](#) (by restriction)

Sub-types: None

Name AreaOfInterestEnum

Content

- Base XSD Type: string
- value comes from list: {'continentWide'|'national'|'neighbouringCountries'|'notSpecified'|'regional'}

Documentation

Types of areas of interest.

### Schema Component Representation

```
<xs:simpleType name="AreaOfInterestEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="continentWide"/>
    <xs:enumeration value="national"/>
  </xs:restriction>
</xs:simpleType>
```

```

<xs:enumeration value="neighbouringCountries"/>
<xs:enumeration value="notSpecified"/>
<xs:enumeration value="regional"/>
</xs:restriction>
</xs:simpleType>

```

[top](#)

### Simple Type: AxlesPerHour

Super-types: [xs:nonNegativeInteger](#) < [NonNegativeInteger](#) (by restriction) < **AxlesPerHour** (by restriction)  
 Sub-types: None

**Name** AxlesPerHour  
**Content**

- Base XSD Type: nonNegativeInteger

**Documentation** Vehicle axles per hour.

#### Schema Component Representation

```

<xs:simpleType name="AxlesPerHour">
  <xs:restriction base="D2LogicalModel:NonNegativeInteger"/>
</xs:simpleType>

```

[top](#)

### Simple Type: Boolean

Super-types: [xs:boolean](#) < **Boolean** (by restriction)  
 Sub-types: None

**Name** Boolean  
**Content**

- Base XSD Type: boolean

**Documentation** Boolean has the value space required to support the mathematical concept of binary-valued logic: {true, false}.

#### Schema Component Representation

```

<xs:simpleType name="Boolean">
  <xs:restriction base="xs:boolean"/>
</xs:simpleType>

```

[top](#)

### Simple Type: CarriagewayEnum

Super-types: [xs:string](#) < **CarriagewayEnum** (by restriction)  
 Sub-types: None

**Name** CarriagewayEnum  
**Content**

- Base XSD Type: string
- value comes from list: {connectingCarriageway|entrySlipRoad|exitSlipRoad|flyover|leftHandFeederRoad|leftHandParallelCarriageway|mainCarriageway|oppositeCarriageway|parallelCarriageway|rightHandFeederRoad|rightHandParallelCarriageway|roundabout|serviceRoad|slipRoads|underpass}

**Documentation** List of descriptors identifying specific carriageway details.

#### Schema Component Representation

```

<xs:simpleType name="CarriagewayEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="connectingCarriageway"/>
    <xs:enumeration value="entrySlipRoad"/>
    <xs:enumeration value="exitSlipRoad"/>
    <xs:enumeration value="flyover"/>
    <xs:enumeration value="leftHandFeederRoad"/>
    <xs:enumeration value="leftHandParallelCarriageway"/>
    <xs:enumeration value="mainCarriageway"/>
    <xs:enumeration value="oppositeCarriageway"/>
    <xs:enumeration value="parallelCarriageway"/>
    <xs:enumeration value="rightHandFeederRoad"/>
    <xs:enumeration value="rightHandParallelCarriageway"/>
    <xs:enumeration value="roundabout"/>
    <xs:enumeration value="serviceRoad"/>
    <xs:enumeration value="slipRoads"/>
    <xs:enumeration value="underpass"/>
  </xs:restriction>
</xs:simpleType>

```

[top](#)

### Simple Type: CodedReasonForSettingMessageEnum

Super-types: [xs:string](#) < **CodedReasonForSettingMessageEnum** (by restriction)  
 Sub-types: None

**Name** CodedReasonForSettingMessageEnum

<b>Content</b>	<ul style="list-style-type: none"> <li>• Base XSD Type: string</li> <li>• <i>value</i> comes from list: {situation operatorCreated trafficManagement travelTime campaign default}</li> </ul>
<b>Documentation</b>	Coded reasons why a message has been selected for display on the sign.

#### Schema Component Representation

```
<xs:simpleType name="CodedReasonForSettingMessageEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="situation"/>
    <xs:enumeration value="operatorCreated"/>
    <xs:enumeration value="trafficManagement"/>
    <xs:enumeration value="travelTime"/>
    <xs:enumeration value="campaign"/>
    <xs:enumeration value="default"/>
  </xs:restriction>
</xs:simpleType>
```

[top](#)

### Simple Type: ColourEnum

Super-types: [xs:string](#) < **ColourEnum** (by restriction)  
 Sub-types: None

<b>Name</b>	ColourEnum
<b>Content</b>	<ul style="list-style-type: none"> <li>• Base XSD Type: string</li> <li>• <i>value</i> comes from list: {amber blue green red white whiteYellow}</li> </ul>
<b>Documentation</b>	Colours.

#### Schema Component Representation

```
<xs:simpleType name="ColourEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="amber"/>
    <xs:enumeration value="blue"/>
    <xs:enumeration value="green"/>
    <xs:enumeration value="red"/>
    <xs:enumeration value="white"/>
    <xs:enumeration value="whiteYellow"/>
  </xs:restriction>
</xs:simpleType>
```

[top](#)

### Simple Type: ComputationMethodEnum

Super-types: [xs:string](#) < **ComputationMethodEnum** (by restriction)  
 Sub-types: None

<b>Name</b>	ComputationMethodEnum
<b>Content</b>	<ul style="list-style-type: none"> <li>• Base XSD Type: string</li> <li>• <i>value</i> comes from list: {arithmeticAverageOfSamplesBasedOnAFixedNumberOfSamples arithmeticAverageOfSamplesInATimePeriod harmonicAverageOfSamplesInATimePeriod}</li> </ul>
<b>Documentation</b>	Types of computational methods used in deriving data values for data sets.

#### Schema Component Representation

```
<xs:simpleType name="ComputationMethodEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="arithmeticAverageOfSamplesBasedOnAFixedNumberOfSamples"/>
    <xs:enumeration value="arithmeticAverageOfSamplesInATimePeriod"/>
    <xs:enumeration value="harmonicAverageOfSamplesInATimePeriod"/>
    <xs:enumeration value="medianOfSamplesInATimePeriod"/>
    <xs:enumeration value="movingAverageOfSamples"/>
  </xs:restriction>
</xs:simpleType>
```

[top](#)

### Simple Type: ConcentrationVehiclesPerKilometre

Super-types: [xs:nonNegativeInteger](#) < [NonNegativeInteger](#) (by restriction) < **ConcentrationVehiclesPerKilometre** (by restriction)  
 Sub-types: None

<b>Name</b>	ConcentrationVehiclesPerKilometre
<b>Content</b>	<ul style="list-style-type: none"> <li>• Base XSD Type: nonNegativeInteger</li> </ul>
<b>Documentation</b>	A measure of traffic density defined in number of vehicles per kilometre of road.

#### Schema Component Representation

```
<xs:simpleType name="ConcentrationVehiclesPerKilometre">
  <xs:restriction base="D2LogicalModel:NonNegativeInteger"/>
</xs:simpleType>
```

## Simple Type: ConfidentialityValueEnum

Super-types:	<a href="#">xs:string</a> < <b>ConfidentialityValueEnum</b> (by restriction)
Sub-types:	None

**Name** ConfidentialityValueEnum

### Content

- Base XSD Type: string
- *value* comes from list:  
{'internalUse'|'noRestriction'|'restrictedToAuthorities'|'restrictedToAuthoritiesAndTrafficOperators'|'restrictedToAuthoritiesTrafficOperatorsAndPublishers'}

**Documentation** Values of confidentiality.

### Schema Component Representation

```
<xs:simpleType name="ConfidentialityValueEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="internalUse"/>
    <xs:enumeration value="noRestriction"/>
    <xs:enumeration value="restrictedToAuthorities"/>
    <xs:enumeration value="restrictedToAuthoritiesAndTrafficOperators"/>
    <xs:enumeration value="restrictedToAuthoritiesTrafficOperatorsAndPublishers"/>
    <xs:enumeration value="restrictedToAuthoritiesTrafficOperatorsAndVms"/>
  </xs:restriction>
</xs:simpleType>
```

## Simple Type: CountryEnum

Super-types:	<a href="#">xs:string</a> < <b>CountryEnum</b> (by restriction)
Sub-types:	None

**Name** CountryEnum

### Content

- Base XSD Type: string
- *value* comes from list:  
{'at'|'be'|'bg'|'ch'|'cs'|'cy'|'cz'|'de'|'dk'|'ee'|'es'|'fi'|'fo'|'fr'|'gb'|'gg'|'gi'|'gr'|'hr'|'hu'|'ie'|'im'|'is'|'it'|'je'|'li'|'lt'|'lu'|'lv'|'ma'|'mc'|'mk'|'mt'|'nl'|'no'|'pl'|'pt'|'ro'|'se'|'si'|'sk'|'sr'}

**Documentation** List of countries.

### Schema Component Representation

```
<xs:simpleType name="CountryEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="at"/>
    <xs:enumeration value="be"/>
    <xs:enumeration value="bg"/>
    <xs:enumeration value="ch"/>
    <xs:enumeration value="cs"/>
    <xs:enumeration value="cy"/>
    <xs:enumeration value="cz"/>
    <xs:enumeration value="de"/>
    <xs:enumeration value="dk"/>
    <xs:enumeration value="ee"/>
    <xs:enumeration value="es"/>
    <xs:enumeration value="fi"/>
    <xs:enumeration value="fo"/>
    <xs:enumeration value="fr"/>
    <xs:enumeration value="gb"/>
    <xs:enumeration value="gg"/>
    <xs:enumeration value="gi"/>
    <xs:enumeration value="gr"/>
    <xs:enumeration value="hr"/>
    <xs:enumeration value="hu"/>
    <xs:enumeration value="ie"/>
    <xs:enumeration value="im"/>
    <xs:enumeration value="is"/>
    <xs:enumeration value="it"/>
    <xs:enumeration value="je"/>
    <xs:enumeration value="li"/>
    <xs:enumeration value="lt"/>
    <xs:enumeration value="lu"/>
    <xs:enumeration value="lv"/>
    <xs:enumeration value="ma"/>
    <xs:enumeration value="mc"/>
    <xs:enumeration value="mk"/>
    <xs:enumeration value="mt"/>
    <xs:enumeration value="nl"/>
    <xs:enumeration value="no"/>
    <xs:enumeration value="pl"/>
    <xs:enumeration value="pt"/>
    <xs:enumeration value="ro"/>
    <xs:enumeration value="se"/>
    <xs:enumeration value="si"/>
    <xs:enumeration value="sk"/>
    <xs:enumeration value="sm"/>
    <xs:enumeration value="tx"/>
    <xs:enumeration value="va"/>
    <xs:enumeration value="other"/>
  </xs:restriction>
</xs:simpleType>
```

## Simple Type: **DateTime**

**Super-types:** [xs:dateTime](#) < **DateTime** (by restriction)  
**Sub-types:** None

**Name** DateTime

### Content

- Base XSD Type: dateTime

### Documentation

A combination of integer-valued year, month, day, hour, minute properties, a decimal-valued second property and a time zone property from which it is possible to determine the local time, the equivalent UTC time and the time zone offset from UTC.

### Schema Component Representation

```
<xs:simpleType name="DateTime">  
  <xs:restriction base="xs:dateTime"/>  
</xs:simpleType>
```

[top](#)

## Simple Type: **DirectionEnum**

**Super-types:** [xs:string](#) < **DirectionEnum** (by restriction)  
**Sub-types:** None

**Name** DirectionEnum

### Content

- Base XSD Type: string
- *value* comes from list:  
{allDirections|bothWays|clockwise|anticlockwise|innerRing|outerRing|northBound|northEastBound|eastBound|southEastBound|southBound}

**Documentation** List of directions of travel.

### Schema Component Representation

```
<xs:simpleType name="DirectionEnum">  
  <xs:restriction base="xs:string">  
    <xs:enumeration value="allDirections"/>  
    <xs:enumeration value="bothWays"/>  
    <xs:enumeration value="clockwise"/>  
    <xs:enumeration value="anticlockwise"/>  
    <xs:enumeration value="innerRing"/>  
    <xs:enumeration value="outerRing"/>  
    <xs:enumeration value="northBound"/>  
    <xs:enumeration value="northEastBound"/>  
    <xs:enumeration value="eastBound"/>  
    <xs:enumeration value="southEastBound"/>  
    <xs:enumeration value="southBound"/>  
    <xs:enumeration value="southWestBound"/>  
    <xs:enumeration value="westBound"/>  
    <xs:enumeration value="northWestBound"/>  
    <xs:enumeration value="inboundTowardsTown"/>  
    <xs:enumeration value="outboundFromTown"/>  
    <xs:enumeration value="unknown"/>  
    <xs:enumeration value="opposite"/>  
    <xs:enumeration value="other"/>  
  </xs:restriction>  
</xs:simpleType>
```

[top](#)

## Simple Type: **ElaboratedDataFaultEnum**

**Super-types:** [xs:string](#) < **ElaboratedDataFaultEnum** (by restriction)  
**Sub-types:** None

**Name** ElaboratedDataFaultEnum

### Content

- Base XSD Type: string
- *value* comes from list:  
{intermittentDataValues|noDataValuesAvailable|spuriousUnreliableDataValues|unspecifiedOrUnknownFault|other}

**Documentation** Types of elaborated data faults.

### Schema Component Representation

```
<xs:simpleType name="ElaboratedDataFaultEnum">  
  <xs:restriction base="xs:string">  
    <xs:enumeration value="intermittentDataValues"/>  
    <xs:enumeration value="noDataValuesAvailable"/>  
    <xs:enumeration value="spuriousUnreliableDataValues"/>  
    <xs:enumeration value="unspecifiedOrUnknownFault"/>  
    <xs:enumeration value="other"/>  
  </xs:restriction>  
</xs:simpleType>
```

[top](#)

## Simple Type: **FaultSeverityEnum**

Super-types: [xs:string](#) < **FaultSeverityEnum** (by restriction)

Sub-types: None

Name FaultSeverityEnum

Content

- Base XSD Type: string
- *value* comes from list: {'low'|'medium'|'high'|'unknown'}

Documentation Classification of the severity of faults.

#### Schema Component Representation

```
<xs:simpleType name="FaultSeverityEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="low"/>
    <xs:enumeration value="medium"/>
    <xs:enumeration value="high"/>
    <xs:enumeration value="unknown"/>
  </xs:restriction>
</xs:simpleType>
```

[top](#)

## Simple Type: **Float**

Super-types: [xs:float](#) < **Float** (by restriction)

Sub-types:

- [KilometresPerHour](#) (by restriction)
- [MetresAsFloat](#) (by restriction)
- [Percentage](#) (by restriction)
- [Seconds](#) (by restriction)
- [Tonnes](#) (by restriction)

Name Float

Content

- Base XSD Type: float

Documentation A floating point number whose value space consists of the values  $m \times 2^e$ , where  $m$  is an integer whose absolute value is less than  $2^{24}$ , and  $e$  is an integer between -149 and 104, inclusive.

#### Schema Component Representation

```
<xs:simpleType name="Float">
  <xs:restriction base="xs:float"/>
</xs:simpleType>
```

[top](#)

## Simple Type: **HeightGradeEnum**

Super-types: [xs:string](#) < **HeightGradeEnum** (by restriction)

Sub-types: None

Name HeightGradeEnum

Content

- Base XSD Type: string
- *value* comes from list: {'aboveGrade'|'atGrade'|'belowGrade'}

Documentation List of height or vertical gradings of road sections.

#### Schema Component Representation

```
<xs:simpleType name="HeightGradeEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="aboveGrade"/>
    <xs:enumeration value="atGrade"/>
    <xs:enumeration value="belowGrade"/>
  </xs:restriction>
</xs:simpleType>
```

[top](#)

## Simple Type: **InformationStatusEnum**

Super-types: [xs:string](#) < **InformationStatusEnum** (by restriction)

Sub-types: None

Name InformationStatusEnum

Content

- Base XSD Type: string
- *value* comes from list: {'real'|'securityExercise'|'technicalExercise'|'test'}

Documentation Status of the related information (i.e. real, test or exercise).

#### Schema Component Representation

```
<xs:simpleType name="InformationStatusEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="real"/>
    <xs:enumeration value="securityExercise"/>
    <xs:enumeration value="technicalExercise"/>
  </xs:restriction>
</xs:simpleType>
```



```
<xs:enumeration value="test"/>
</xs:restriction>
</xs:simpleType>
```

[top](#)

## Simple Type: Integer

Super-types: [xs:integer](#) < **Integer** (by restriction)  
Sub-types: None

Name Integer

Content

- Base XSD Type: integer

Documentation An integer number whose value space is the set {-2147483648, -2147483647, -2147483646, ..., -2, -1, 0, 1, 2, ..., 2147483645, 2147483646, 2147483647}.

### Schema Component Representation

```
<xs:simpleType name="Integer">
  <xs:restriction base="xs:integer"/>
</xs:simpleType>
```

[top](#)

## Simple Type: JunctionClassificationEnum

Super-types: [xs:string](#) < **JunctionClassificationEnum** (by restriction)  
Sub-types: None

Name JunctionClassificationEnum

Content

- Base XSD Type: string
- *value* comes from list: {threeWayInterchange|interchange|motorwayConnection|junction|temporaryJunction|borderCrossing|junctionInOneDirection|operationalServiceJunction|other}

Documentation Explicit type of a junction.

### Schema Component Representation

```
<xs:simpleType name="JunctionClassificationEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="threeWayInterchange"/>
    <xs:enumeration value="interchange"/>
    <xs:enumeration value="motorwayConnection"/>
    <xs:enumeration value="junction"/>
    <xs:enumeration value="temporaryJunction"/>
    <xs:enumeration value="borderCrossing"/>
    <xs:enumeration value="junctionInOneDirection"/>
    <xs:enumeration value="operationalServiceJunction"/>
    <xs:enumeration value="other"/>
  </xs:restriction>
</xs:simpleType>
```

[top](#)

## Simple Type: KilometresPerHour

Super-types: [xs:float](#) < [Float](#) (by restriction) < **KilometresPerHour** (by restriction)  
Sub-types: None

Name KilometresPerHour

Content

- Base XSD Type: float

Documentation A measure of speed defined in kilometres per hour.

### Schema Component Representation

```
<xs:simpleType name="KilometresPerHour">
  <xs:restriction base="D2LogicalModel:Float"/>
</xs:simpleType>
```

[top](#)

## Simple Type: LaneEnum

Super-types: [xs:string](#) < **LaneEnum** (by restriction)  
Sub-types: None

Name LaneEnum

Content

- Base XSD Type: string
- *value* comes from list: {emergencyLane|leftLane|middleLane|rightLane}

Documentation List of descriptors identifying specific lanes.

### Schema Component Representation

```

<xs:simpleType name="LaneEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="emergencyLane"/>
    <xs:enumeration value="leftLane"/>
    <xs:enumeration value="middleLane"/>
    <xs:enumeration value="rightLane"/>
  </xs:restriction>
</xs:simpleType>

```

[top](#)

## Simple Type: Language

Super-types: [xs:language](#) < **Language** (by restriction)

Sub-types: None

**Name** Language

**Content**

- Base XSD Type: language

**Documentation** A language datatype, identifies a specified language by an ISO 639-1 2-alpha / ISO 639-2 3-alpha code.

### Schema Component Representation

```

<xs:simpleType name="Language">
  <xs:restriction base="xs:language"/>
</xs:simpleType>

```

[top](#)

## Simple Type: LinearElementNatureEnum

Super-types: [xs:string](#) < **LinearElementNatureEnum** (by restriction)

Sub-types: None

**Name** LinearElementNatureEnum

**Content**

- Base XSD Type: string
- *value* comes from list: {'road'|'roadSection'|'slipRoad'|'other'}

**Documentation** List of indicative natures of linear elements.

### Schema Component Representation

```

<xs:simpleType name="LinearElementNatureEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="road"/>
    <xs:enumeration value="roadSection"/>
    <xs:enumeration value="slipRoad"/>
    <xs:enumeration value="other"/>
  </xs:restriction>
</xs:simpleType>

```

[top](#)

## Simple Type: LinearReferencingDirectionEnum

Super-types: [xs:string](#) < **LinearReferencingDirectionEnum** (by restriction)

Sub-types: None

**Name** LinearReferencingDirectionEnum

**Content**

- Base XSD Type: string
- *value* comes from list: {'both'|'opposite'|'aligned'|'unknown'}

**Documentation** Directions of traffic flow relative to the direction in which the linear element is defined.

### Schema Component Representation

```

<xs:simpleType name="LinearReferencingDirectionEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="both"/>
    <xs:enumeration value="opposite"/>
    <xs:enumeration value="aligned"/>
    <xs:enumeration value="unknown"/>
  </xs:restriction>
</xs:simpleType>

```

[top](#)

## Simple Type: LocationDescriptorEnum

Super-types: [xs:string](#) < **LocationDescriptorEnum** (by restriction)

Sub-types: None

**Name** LocationDescriptorEnum

**Content**

- Base XSD Type: string

- *value* comes from list:  
{'aroundABendInRoad'|'atMotorwayInterchange'|'atRestArea'|'atServiceArea'|'atTollPlaza'|'atTunnelEntryOrExit'|'inbound'|'inGallery'|'inTheCentre'|'inT

**Documentation** List of descriptors to help to identify a specific location.

#### Schema Component Representation

```
<xs:simpleType name="LocationDescriptorEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="aroundABendInRoad"/>
    <xs:enumeration value="atMotorwayInterchange"/>
    <xs:enumeration value="atRestArea"/>
    <xs:enumeration value="atServiceArea"/>
    <xs:enumeration value="atTollPlaza"/>
    <xs:enumeration value="atTunnelEntryOrExit"/>
    <xs:enumeration value="inbound"/>
    <xs:enumeration value="inGallery"/>
    <xs:enumeration value="inTheCentre"/>
    <xs:enumeration value="inTheOppositeDirection"/>
    <xs:enumeration value="inTunnel"/>
    <xs:enumeration value="onBorder"/>
    <xs:enumeration value="onBridge"/>
    <xs:enumeration value="onConnector"/>
    <xs:enumeration value="onElevatedSection"/>
    <xs:enumeration value="onFlyover"/>
    <xs:enumeration value="onIceRoad"/>
    <xs:enumeration value="onLevelCrossing"/>
    <xs:enumeration value="onLinkRoad"/>
    <xs:enumeration value="onPass"/>
    <xs:enumeration value="onRoundabout"/>
    <xs:enumeration value="onTheLeft"/>
    <xs:enumeration value="onTheRight"/>
    <xs:enumeration value="onTheRoadway"/>
    <xs:enumeration value="onUndergroundSection"/>
    <xs:enumeration value="onUnderpass"/>
    <xs:enumeration value="outbound"/>
    <xs:enumeration value="overCrestOfHill"/>
    <xs:enumeration value="withinJunction"/>
  </xs:restriction>
</xs:simpleType>
```

[top](#)

### Simple Type: MeasurementEquipmentFaultEnum

**Super-types:** [xs:string](#) < **MeasurementEquipmentFaultEnum** (by restriction)

**Sub-types:** None

**Name** MeasurementEquipmentFaultEnum

**Content**

- Base XSD Type: string
- *value* comes from list:  
{'intermittentDataValues'|'noDataValuesAvailable'|'spuriousUnreliableDataValues'|'unspecifiedOrUnknownFault'|'other'}

**Documentation** Types of measurement equipment faults.

#### Schema Component Representation

```
<xs:simpleType name="MeasurementEquipmentFaultEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="intermittentDataValues"/>
    <xs:enumeration value="noDataValuesAvailable"/>
    <xs:enumeration value="spuriousUnreliableDataValues"/>
    <xs:enumeration value="unspecifiedOrUnknownFault"/>
    <xs:enumeration value="other"/>
  </xs:restriction>
</xs:simpleType>
```

[top](#)

### Simple Type: MetresAsFloat

**Super-types:** [xs:float](#) < [Float](#) (by restriction) < **MetresAsFloat** (by restriction)

**Sub-types:** None

**Name** MetresAsFloat

**Content**

- Base XSD Type: float

**Documentation** A measure of distance defined in metres in a floating point format.

#### Schema Component Representation

```
<xs:simpleType name="MetresAsFloat">
  <xs:restriction base="D2LogicalModel:Float"/>
</xs:simpleType>
```

[top](#)

### Simple Type: MetresAsNonNegativeInteger

**Super-types:** [xs:nonNegativeInteger](#) < [NonNegativeInteger](#) (by restriction) < **MetresAsNonNegativeInteger** (by restriction)

**Sub-types:** None

<b>Name</b>	MetresAsNonNegativeInteger
<b>Content</b>	<ul style="list-style-type: none"> <li>• Base XSD Type: nonNegativeInteger</li> </ul>
<b>Documentation</b>	A measure of distance defined in metres in a non negative integer format.

#### Schema Component Representation

```
<xs:simpleType name="MetresAsNonNegativeInteger">
  <xs:restriction base="D2LogicalModel:NonNegativeInteger" />
</xs:simpleType>
```

[top](#)

### Simple Type: MultilingualStringValue

<b>Super-types:</b>	<a href="#">xs:string</a> < <b>MultilingualStringValue</b> (by restriction)
<b>Sub-types:</b>	<ul style="list-style-type: none"> <li>• <a href="#">MultilingualStringValue</a> (by extension)</li> </ul>

<b>Name</b>	MultilingualStringValue
<b>Content</b>	<ul style="list-style-type: none"> <li>• Base XSD Type: string</li> <li>• <i>length</i> &lt;= 1024</li> </ul>

#### Schema Component Representation

```
<xs:simpleType name="MultilingualStringValue">
  <xs:restriction base="xs:string">
    <xs:maxLength value="1024" />
  </xs:restriction>
</xs:simpleType>
```

[top](#)

### Simple Type: NonNegativeInteger

<b>Super-types:</b>	<a href="#">xs:nonNegativeInteger</a> < <b>NonNegativeInteger</b> (by restriction)
<b>Sub-types:</b>	<ul style="list-style-type: none"> <li>• <a href="#">AlertCLocationCode</a> (by restriction)</li> <li>• <a href="#">AngleInDegrees</a> (by restriction)</li> <li>• <a href="#">AxlesPerHour</a> (by restriction)</li> <li>• <a href="#">ConcentrationVehiclesPerKilometre</a> (by restriction)</li> <li>• <a href="#">MetresAsNonNegativeInteger</a> (by restriction)</li> <li>• <a href="#">PassengerCarUnitsPerHour</a> (by restriction)</li> <li>• <a href="#">VehiclesPerHour</a> (by restriction)</li> </ul>

<b>Name</b>	NonNegativeInteger
<b>Content</b>	<ul style="list-style-type: none"> <li>• Base XSD Type: nonNegativeInteger</li> </ul>
<b>Documentation</b>	An integer number whose value space is the set {0, 1, 2, ..., 2147483645, 2147483646, 2147483647}.

#### Schema Component Representation

```
<xs:simpleType name="NonNegativeInteger">
  <xs:restriction base="xs:nonNegativeInteger" />
</xs:simpleType>
```

[top](#)

### Simple Type: OpenIrFormOfWayEnum

<b>Super-types:</b>	<a href="#">xs:string</a> < <b>OpenIrFormOfWayEnum</b> (by restriction)
<b>Sub-types:</b>	None

<b>Name</b>	OpenIrFormOfWayEnum
<b>Content</b>	<ul style="list-style-type: none"> <li>• Base XSD Type: string</li> <li>• <i>value</i> comes from list: {undefined 'motorway' 'multipleCarriageway' 'singleCarriageway' 'roundabout' 'slipRoad' 'trafficSquare' 'other'}</li> </ul>
<b>Documentation</b>	Enumeration of for of way

#### Schema Component Representation

```
<xs:simpleType name="OpenIrFormOfWayEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="undefined" />
    <xs:enumeration value="motorway" />
    <xs:enumeration value="multipleCarriageway" />
    <xs:enumeration value="singleCarriageway" />
    <xs:enumeration value="roundabout" />
    <xs:enumeration value="slipRoad" />
    <xs:enumeration value="trafficSquare" />
    <xs:enumeration value="other" />
  </xs:restriction>
</xs:simpleType>
```

[top](#)

### Simple Type: OpenIrFunctionalRoadClassEnum

**Super-types:** [xs:string](#) < **OpenlrFunctionalRoadClassEnum** (by restriction)  
**Sub-types:** None

**Name** OpenlrFunctionalRoadClassEnum

**Content**

- Base XSD Type: string
- *value* comes from list: {'FRC0'|'FRC1'|'FRC2'|'FRC3'|'FRC4'|'FRC5'|'FRC6'|'FRC7'}

**Documentation** Enumeration of functional road class

#### Schema Component Representation

```
<xs:simpleType name="OpenlrFunctionalRoadClassEnum">  
  <xs:restriction base="xs:string">  
    <xs:enumeration value="FRC0"/>  
    <xs:enumeration value="FRC1"/>  
    <xs:enumeration value="FRC2"/>  
    <xs:enumeration value="FRC3"/>  
    <xs:enumeration value="FRC4"/>  
    <xs:enumeration value="FRC5"/>  
    <xs:enumeration value="FRC6"/>  
    <xs:enumeration value="FRC7"/>  
  </xs:restriction>  
</xs:simpleType>
```

[top](#)

### Simple Type: **OpenlrOrientationEnum**

**Super-types:** [xs:string](#) < **OpenlrOrientationEnum** (by restriction)  
**Sub-types:** None

**Name** OpenlrOrientationEnum

**Content**

- Base XSD Type: string
- *value* comes from list: {'noOrientationOrUnknown'|'withLineDirection'|'againstLineDirection'|'both'}

**Documentation** Enumeration of side of road

#### Schema Component Representation

```
<xs:simpleType name="OpenlrOrientationEnum">  
  <xs:restriction base="xs:string">  
    <xs:enumeration value="noOrientationOrUnknown"/>  
    <xs:enumeration value="withLineDirection"/>  
    <xs:enumeration value="againstLineDirection"/>  
    <xs:enumeration value="both"/>  
  </xs:restriction>  
</xs:simpleType>
```

[top](#)

### Simple Type: **OpenlrSideOfRoadEnum**

**Super-types:** [xs:string](#) < **OpenlrSideOfRoadEnum** (by restriction)  
**Sub-types:** None

**Name** OpenlrSideOfRoadEnum

**Content**

- Base XSD Type: string
- *value* comes from list: {'onRoadOrUnknown'|'right'|'left'|'both'}

**Documentation** Enumeration of side of road

#### Schema Component Representation

```
<xs:simpleType name="OpenlrSideOfRoadEnum">  
  <xs:restriction base="xs:string">  
    <xs:enumeration value="onRoadOrUnknown"/>  
    <xs:enumeration value="right"/>  
    <xs:enumeration value="left"/>  
    <xs:enumeration value="both"/>  
  </xs:restriction>  
</xs:simpleType>
```

[top](#)

### Simple Type: **PassengerCarUnitsPerHour**

**Super-types:** [xs:nonNegativeInteger](#) < [NonNegativeInteger](#) (by restriction) < **PassengerCarUnitsPerHour** (by restriction)  
**Sub-types:** None

**Name** PassengerCarUnitsPerHour

**Content**

- Base XSD Type: nonNegativeInteger

**Documentation** Passenger car units per hour.

#### Schema Component Representation

```
<xs:simpleType name="PassengerCarUnitsPerHour">
  <xs:restriction base="D2LogicalModel:NonNegativeInteger"/>
</xs:simpleType>
```

[top](#)

## Simple Type: Percentage

Super-types: [xs:float](#) < [Float](#) (by restriction) < **Percentage** (by restriction)  
Sub-types: None

**Name** Percentage  
**Content**

- Base XSD Type: float

**Documentation** A measure of percentage.

### Schema Component Representation

```
<xs:simpleType name="Percentage">
  <xs:restriction base="D2LogicalModel:Float"/>
</xs:simpleType>
```

[top](#)

## Simple Type: ReferentTypeEnum

Super-types: [xs:string](#) < **ReferentTypeEnum** (by restriction)  
Sub-types: None

**Name** ReferentTypeEnum  
**Content**

- Base XSD Type: string
- *value* comes from list: {'boundary'|'intersection'|'referenceMarker'|'landmark'|'roadNode'}

**Documentation** A set of types of known points along a linear object such as a road.

### Schema Component Representation

```
<xs:simpleType name="ReferentTypeEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="boundary"/>
    <xs:enumeration value="intersection"/>
    <xs:enumeration value="referenceMarker"/>
    <xs:enumeration value="landmark"/>
    <xs:enumeration value="roadNode"/>
  </xs:restriction>
</xs:simpleType>
```

[top](#)

## Simple Type: RoadTypeEnum

Super-types: [xs:string](#) < **RoadTypeEnum** (by restriction)  
Sub-types: None

**Name** RoadTypeEnum  
**Content**

- Base XSD Type: string
- *value* comes from list: {'motorway'|'trunkRoad'|'mainRoad'|'other'}

**Documentation** Categorisation of the road type (motorway, main road, ...).

### Schema Component Representation

```
<xs:simpleType name="RoadTypeEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="motorway"/>
    <xs:enumeration value="trunkRoad"/>
    <xs:enumeration value="mainRoad"/>
    <xs:enumeration value="other"/>
  </xs:restriction>
</xs:simpleType>
```

[top](#)

## Simple Type: Seconds

Super-types: [xs:float](#) < [Float](#) (by restriction) < **Seconds** (by restriction)  
Sub-types: None

**Name** Seconds  
**Content**

- Base XSD Type: float

**Documentation** Seconds.

### Schema Component Representation

```
<xs:simpleType name="Seconds">
  <xs:restriction base="D2LogicalModel:Float"/>
</xs:simpleType>
```

[top](#)

## Simple Type: String

**Super-types:** [xs:string](#) < **String** (by restriction)  
**Sub-types:** None

**Name** String  
**Content**

- Base XSD Type: string
- *length* <= 1024

**Documentation** A character string whose value space is the set of finite-length sequences of characters. Every character has a corresponding Universal Character Set code point (as defined in ISO/IEC 10646), which is an integer.

### Schema Component Representation

```
<xs:simpleType name="String">
  <xs:restriction base="xs:string">
    <xs:maxLength value="1024"/>
  </xs:restriction>
</xs:simpleType>
```

[top](#)

## Simple Type: Tonnes

**Super-types:** [xs:float](#) < [Float](#) (by restriction) < **Tonnes** (by restriction)  
**Sub-types:** None

**Name** Tonnes  
**Content**

- Base XSD Type: float

**Documentation** A measure of weight defined in metric tonnes.

### Schema Component Representation

```
<xs:simpleType name="Tonnes">
  <xs:restriction base="D2LogicalModel:Float"/>
</xs:simpleType>
```

[top](#)

## Simple Type: TpegLoc01FramedPointLocationSubtypeEnum

**Super-types:** [xs:string](#) < **TpegLoc01FramedPointLocationSubtypeEnum** (by restriction)  
**Sub-types:** None

**Name** TpegLoc01FramedPointLocationSubtypeEnum  
**Content**

- Base XSD Type: string
- *value* comes from list: {'framedPoint'}

**Documentation** Types of points on the road network framed by two other points on the same road.

### Schema Component Representation

```
<xs:simpleType name="TpegLoc01FramedPointLocationSubtypeEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="framedPoint"/>
  </xs:restriction>
</xs:simpleType>
```

[top](#)

## Simple Type: TpegLoc01LinearLocationSubtypeEnum

**Super-types:** [xs:string](#) < **TpegLoc01LinearLocationSubtypeEnum** (by restriction)  
**Sub-types:** None

**Name** TpegLoc01LinearLocationSubtypeEnum  
**Content**

- Base XSD Type: string
- *value* comes from list: {'segment'}

**Documentation** Types of linear location.

### Schema Component Representation

```
<xs:simpleType name="TpegLoc01LinearLocationSubtypeEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="segment"/>
  </xs:restriction>
</xs:simpleType>
```



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

[top](#)

### Simple Type: TpegLoc01SimplePointLocationSubtypeEnum

**Super-types:** [xs:string](#) < **TpegLoc01SimplePointLocationSubtypeEnum** (by restriction)  
**Sub-types:** None

**Name** TpegLoc01SimplePointLocationSubtypeEnum  
**Content**

- Base XSD Type: string
- *value* comes from list: {'intersection'|'nonLinkedPoint'}

**Documentation** Types of simple point.

#### Schema Component Representation

```
<xs:simpleType name="TpegLoc01SimplePointLocationSubtypeEnum">  
  <xs:restriction base="xs:string">  
    <xs:enumeration value="intersection"/>  
    <xs:enumeration value="nonLinkedPoint"/>  
  </xs:restriction>  
</xs:simpleType>
```

[top](#)

### Simple Type: TpegLoc03IlcPointDescriptorSubtypeEnum

**Super-types:** [xs:string](#) < **TpegLoc03IlcPointDescriptorSubtypeEnum** (by restriction)  
**Sub-types:** None

**Name** TpegLoc03IlcPointDescriptorSubtypeEnum  
**Content**

- Base XSD Type: string
- *value* comes from list: {'tpegIlcName1'|'tpegIlcName2'|'tpegIlcName3'}

**Documentation** Descriptors for describing a junction by identifying the intersecting roads at a road junction.

#### Schema Component Representation

```
<xs:simpleType name="TpegLoc03IlcPointDescriptorSubtypeEnum">  
  <xs:restriction base="xs:string">  
    <xs:enumeration value="tpegIlcName1"/>  
    <xs:enumeration value="tpegIlcName2"/>  
    <xs:enumeration value="tpegIlcName3"/>  
  </xs:restriction>  
</xs:simpleType>
```

[top](#)

### Simple Type: TpegLoc03JunctionPointDescriptorSubtypeEnum

**Super-types:** [xs:string](#) < **TpegLoc03JunctionPointDescriptorSubtypeEnum** (by restriction)  
**Sub-types:** None

**Name** TpegLoc03JunctionPointDescriptorSubtypeEnum  
**Content**

- Base XSD Type: string
- *value* comes from list: {'junctionName'}

**Documentation** Descriptors for describing a point at a road junction.

#### Schema Component Representation

```
<xs:simpleType name="TpegLoc03JunctionPointDescriptorSubtypeEnum">  
  <xs:restriction base="xs:string">  
    <xs:enumeration value="junctionName"/>  
  </xs:restriction>  
</xs:simpleType>
```

[top](#)

### Simple Type: TpegLoc03OtherPointDescriptorSubtypeEnum

**Super-types:** [xs:string](#) < **TpegLoc03OtherPointDescriptorSubtypeEnum** (by restriction)  
**Sub-types:** None

**Name** TpegLoc03OtherPointDescriptorSubtypeEnum  
**Content**

- Base XSD Type: string
- *value* comes from list: {'administrativeAreaName'|'administrativeReferenceName'|'airportName'|'areaName'|'buildingName'|'busStopIdentifier'|'busStopName'|'canalName'|'c

**Documentation** Descriptors other than junction names and road descriptors which can help to identify the location of points on the road network.

#### Schema Component Representation

```

<xs:simpleType name="TpegLoc030therPointDescriptorSubtypeEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="administrativeAreaName"/>
    <xs:enumeration value="administrativeReferenceName"/>
    <xs:enumeration value="airportName"/>
    <xs:enumeration value="areaName"/>
    <xs:enumeration value="buildingName"/>
    <xs:enumeration value="busStopIdentifier"/>
    <xs:enumeration value="busStopName"/>
    <xs:enumeration value="canalName"/>
    <xs:enumeration value="countyName"/>
    <xs:enumeration value="ferryPortName"/>
    <xs:enumeration value="intersectionName"/>
    <xs:enumeration value="lakeName"/>
    <xs:enumeration value="linkName"/>
    <xs:enumeration value="localLinkName"/>
    <xs:enumeration value="metroStationName"/>
    <xs:enumeration value="nationName"/>
    <xs:enumeration value="nonLinkedPointName"/>
    <xs:enumeration value="parkingFacilityName"/>
    <xs:enumeration value="pointName"/>
    <xs:enumeration value="pointOfInterestName"/>
    <xs:enumeration value="railwayStation"/>
    <xs:enumeration value="regionName"/>
    <xs:enumeration value="riverName"/>
    <xs:enumeration value="seaName"/>
    <xs:enumeration value="serviceAreaName"/>
    <xs:enumeration value="tidalRiverName"/>
    <xs:enumeration value="townName"/>
    <xs:enumeration value="other"/>
  </xs:restriction>
</xs:simpleType>

```

[top](#)

### Simple Type: **TrafficStatusEnum**

Super-types: [xs:string](#) < **TrafficStatusEnum** (by restriction)  
 Sub-types: None

**Name** TrafficStatusEnum  
**Content**

- Base XSD Type: string
- *value* comes from list: {'impossible'|'congested'|'heavy'|'freeFlow'|'unknown'}

**Documentation** List of terms used to describe traffic conditions.

#### Schema Component Representation

```

<xs:simpleType name="TrafficStatusEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="impossible"/>
    <xs:enumeration value="congested"/>
    <xs:enumeration value="heavy"/>
    <xs:enumeration value="freeFlow"/>
    <xs:enumeration value="unknown"/>
  </xs:restriction>
</xs:simpleType>

```

[top](#)

### Simple Type: **UrgencyEnum**

Super-types: [xs:string](#) < **UrgencyEnum** (by restriction)  
 Sub-types: None

**Name** UrgencyEnum  
**Content**

- Base XSD Type: string
- *value* comes from list: {'extremelyUrgent'|'urgent'|'normalUrgency'}

**Documentation** Degrees of urgency that a receiving client should associate with the disseminate of the information contained in the publication.

#### Schema Component Representation

```

<xs:simpleType name="UrgencyEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="extremelyUrgent"/>
    <xs:enumeration value="urgent"/>
    <xs:enumeration value="normalUrgency"/>
  </xs:restriction>
</xs:simpleType>

```

[top](#)

### Simple Type: **Url**

Super-types: [xs:anyURI](#) < **Url** (by restriction)  
 Sub-types: None

**Name** Url

<b>Content</b>	<ul style="list-style-type: none"> <li>• Base XSD Type: anyURI</li> </ul>
<b>Documentation</b>	A Uniform Resource Locator (URL) address comprising a compact string of characters for a resource available on the Internet.

#### Schema Component Representation

```
<xs:simpleType name="Uri">
  <xs:restriction base="xs:anyURI"/>
</xs:simpleType>
```

[top](#)

### Simple Type: **VehiclesPerHour**

<b>Super-types:</b>	<a href="#">xs:nonNegativeInteger</a> < <a href="#">NonNegativeInteger</a> (by restriction) < <b>VehiclesPerHour</b> (by restriction)
<b>Sub-types:</b>	None

<b>Name</b>	VehiclesPerHour
<b>Content</b>	<ul style="list-style-type: none"> <li>• Base XSD Type: nonNegativeInteger</li> </ul>
<b>Documentation</b>	Vehicles per hour.

#### Schema Component Representation

```
<xs:simpleType name="VehiclesPerHour">
  <xs:restriction base="D2LogicalModel:NonNegativeInteger"/>
</xs:simpleType>
```

[top](#)

### Simple Type: **VmsDatexPictogramEnum**

<b>Super-types:</b>	<a href="#">xs:string</a> < <b>VmsDatexPictogramEnum</b> (by restriction)
<b>Sub-types:</b>	None

<b>Name</b>	VmsDatexPictogramEnum
<b>Content</b>	<ul style="list-style-type: none"> <li>• Base XSD Type: string</li> <li>• <i>value</i> comes from list:           <ul style="list-style-type: none"> <li>{'accident' 'advisorySpeed' 'animalsOnRoad' 'blankVoid' 'bridgeClosed' 'bridgeSwingInOperation' 'carParkFull' 'carParkSpacesAvailable' 'carriageway'</li> </ul> </li> </ul>
<b>Documentation</b>	Types of main pictograms.

#### Schema Component Representation

```
<xs:simpleType name="VmsDatexPictogramEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="accident"/>
    <xs:enumeration value="advisorySpeed"/>
    <xs:enumeration value="animalsOnRoad"/>
    <xs:enumeration value="blankVoid"/>
    <xs:enumeration value="bridgeClosed"/>
    <xs:enumeration value="bridgeSwingInOperation"/>
    <xs:enumeration value="carParkFull"/>
    <xs:enumeration value="carParkSpacesAvailable"/>
    <xs:enumeration value="carriagewayNarrows"/>
    <xs:enumeration value="carriagewayNarrowsOnTheLeft"/>
    <xs:enumeration value="carriagewayNarrowsOnTheRight"/>
    <xs:enumeration value="carriagewayReducedToOneLane"/>
    <xs:enumeration value="carriagewayReducedToTwoLanes"/>
    <xs:enumeration value="carriagewayReducedToThreeLanes"/>
    <xs:enumeration value="chainsOrSnowTyresRecommended"/>
    <xs:enumeration value="compulsoryMinimumSpeed"/>
    <xs:enumeration value="crossWind"/>
    <xs:enumeration value="dangerOfFire"/>
    <xs:enumeration value="drivingOfVehiclesLessThanXMetresApartProhibited"/>
    <xs:enumeration value="endOfAdvisorySpeed"/>
    <xs:enumeration value="endOfCompulsoryMinimumSpeed"/>
    <xs:enumeration value="endOfProhibitionOfOvertaking"/>
    <xs:enumeration value="endOfProhibitionOfOvertakingForGoodsVehicles"/>
    <xs:enumeration value="endOfSpeedLimit"/>
    <xs:enumeration value="exitClosed"/>
    <xs:enumeration value="fallingRocks"/>
    <xs:enumeration value="fastenChildrensSeatBelts"/>
    <xs:enumeration value="fastenYourSeatBelt"/>
    <xs:enumeration value="fire"/>
    <xs:enumeration value="floodingOrFlashFloods"/>
    <xs:enumeration value="fog"/>
    <xs:enumeration value="footballMatch"/>
    <xs:enumeration value="hardShoulderNotRunning"/>
    <xs:enumeration value="hardShoulderRunning"/>
    <xs:enumeration value="keepASafeDistance"/>
    <xs:enumeration value="keepLeft"/>
    <xs:enumeration value="keepRight"/>
    <xs:enumeration value="lane1ClosedOf2"/>
    <xs:enumeration value="lane2ClosedOf2"/>
    <xs:enumeration value="lane1ClosedOf3"/>
    <xs:enumeration value="lane3ClosedOf3"/>
    <xs:enumeration value="lanes1And2ClosedOf3"/>
    <xs:enumeration value="lanes2And3ClosedOf3"/>
    <xs:enumeration value="lane1ClosedOf4"/>
    <xs:enumeration value="lane4ClosedOf4"/>
    <xs:enumeration value="lanes1And2ClosedOf4"/>
    <xs:enumeration value="lanes3And4ClosedOf4"/>
    <xs:enumeration value="lanes1And2And3ClosedOf4"/>
    <xs:enumeration value="lanes2And3And4ClosedOf4"/>
  </xs:restriction>
</xs:simpleType>
```

```

<xs:enumeration value="laneClosed"/>
<xs:enumeration value="laneDeviationToLeft"/>
<xs:enumeration value="laneDeviationToRight"/>
<xs:enumeration value="laneOpen"/>
<xs:enumeration value="leftHandLaneClosed"/>
<xs:enumeration value="lightSignals"/>
<xs:enumeration value="looseGravel"/>
<xs:enumeration value="maintenanceVehicleInAction"/>
<xs:enumeration value="maximumSpeedLimitedToTheFigureIndicated"/>
<xs:enumeration value="narrowLanesAhead"/>
<xs:enumeration value="noEntry"/>
<xs:enumeration value="noEntryForAnyPowerDrivenVehicleDrawingATrailer"/>
<xs:enumeration
value="noEntryForAnyPowerDrivenVehicleDrawingATrailerOtherThanASemiTrailerOrASingleAxleTrailer"/>
<xs:enumeration value="noEntryForGoodsVehicles"/>
<xs:enumeration value="noEntryForVehiclesExceedingXTonnesLadenMass"/>
<xs:enumeration value="noEntryForVehiclesHavingAMassExceedingXTonnesOnOneAxle"/>
<xs:enumeration value="noEntryForVehiclesHavingAnOverallHeightExceedingXMetres"/>
<xs:enumeration value="noEntryForVehiclesHavingAnOverallLengthExceedingXMetres"/>
<xs:enumeration value="noEntryForVehiclesHavingAnOverallWidthExceedingXMetres"/>
<xs:enumeration value="noEntryForVehiclesCarryingDangerousGoods"/>
<xs:enumeration value="otherDangers"/>
<xs:enumeration value="overtakingByGoodsVehiclesProhibited"/>
<xs:enumeration value="overtakingProhibited"/>
<xs:enumeration value="pollutionOrSmogAlert"/>
<xs:enumeration value="queue"/>
<xs:enumeration value="rain"/>
<xs:enumeration value="rightHandLaneClosed"/>
<xs:enumeration value="roadClosedAhead"/>
<xs:enumeration value="roadworks"/>
<xs:enumeration value="slipperyRoad"/>
<xs:enumeration value="smoke"/>
<xs:enumeration value="snow"/>
<xs:enumeration value="snowChainsCompulsory"/>
<xs:enumeration value="snowTyresCompulsory"/>
<xs:enumeration value="snowPloughInAction"/>
<xs:enumeration value="speedCamerasInAction"/>
<xs:enumeration value="trafficCongestion"/>
<xs:enumeration value="trafficDeviatedToOppositeCarriagewayAhead"/>
<xs:enumeration value="trafficPartiallyDeviatedToOppositeCarriagewayAhead"/>
<xs:enumeration value="tunnelClosed"/>
<xs:enumeration value="turnLeft"/>
<xs:enumeration value="turnRight"/>
<xs:enumeration value="twoWayTraffic"/>
<xs:enumeration value="unevenRoad"/>
<xs:enumeration value="vehicleFire"/>
<xs:enumeration value="other"/>
</xs:restriction>
</xs:simpleType>

```

[top](#)

### Simple Type: **VmsDatexSupplementalPictogramEnum**

**Super-types:** [xs:string](#) < **VmsDatexSupplementalPictogramEnum** (by restriction)  
**Sub-types:** None

**Name** VmsDatexSupplementalPictogramEnum

**Content**

- Base XSD Type: string
- *value* comes from list:  
 {distanceToTheBeginningofTheApplicationZone|exceptAnyPowerDrivenVehicleDrawingTrailer|exceptBus|exceptGoodsVehicles|exceptSemiTrail

**Documentation** Types of pictograms displayable in supplementary panels (normally below the main pictogram display which it qualifies).

#### Schema Component Representation

```

<xs:simpleType name="VmsDatexSupplementalPictogramEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="distanceToTheBeginningofTheApplicationZone"/>
    <xs:enumeration value="exceptAnyPowerDrivenVehicleDrawingTrailer"/>
    <xs:enumeration value="exceptBus"/>
    <xs:enumeration value="exceptGoodsVehicles"/>
    <xs:enumeration value="exceptSemiTrailer"/>
    <xs:enumeration value="exceptVehiclesCarryingDangerousGoods"/>
    <xs:enumeration value="inCaseOfIceOrSnow"/>
    <xs:enumeration value="lengthOfTheApplicationZone"/>
    <xs:enumeration value="restrictedToAnyPowerDrivenVehicleDrawingTrailer"/>
    <xs:enumeration value="restrictetdToBus"/>
    <xs:enumeration value="restrictedToGoodsVehicles"/>
    <xs:enumeration value="restrictedToSemiTrailer"/>
    <xs:enumeration value="restrictedToVehiclesCarryingDangerousGoods"/>
    <xs:enumeration value="maintenanceVehicles"/>
    <xs:enumeration value="snowPloughs"/>
    <xs:enumeration value="other"/>
  </xs:restriction>
</xs:simpleType>

```

[top](#)

### Simple Type: **VmsFaultEnum**

**Super-types:** [xs:string](#) < **VmsFaultEnum** (by restriction)  
**Sub-types:** None

**Name** VmsFaultEnum

- Content**
- Base XSD Type: string
  - *value* comes from list:  
{communicationsFailure|incorrectMessageDisplayed|incorrectPictogramDisplayed|outOfService|powerFailure|unableToClearDown|unknown|o'}
- Documentation** Types of variable message sign faults.

#### Schema Component Representation

```
<xs:simpleType name="VmsFaultEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="communicationsFailure"/>
    <xs:enumeration value="incorrectMessageDisplayed"/>
    <xs:enumeration value="incorrectPictogramDisplayed"/>
    <xs:enumeration value="outOfService"/>
    <xs:enumeration value="powerFailure"/>
    <xs:enumeration value="unableToClearDown"/>
    <xs:enumeration value="unknown"/>
    <xs:enumeration value="other"/>
  </xs:restriction>
</xs:simpleType>
```

[top](#)

### Simple Type: VmsLuminanceLevelEnum

**Super-types:** [xs:string](#) < **VmsLuminanceLevelEnum** (by restriction)

**Sub-types:** None

- Name** VmsLuminanceLevelEnum
- Content**
- Base XSD Type: string
  - *value* comes from list:  
{switchedOff|testing|night|overcast|broadDaylight|sunInEyes|sunOnBack|foggyDay|foggyNight}
- Documentation** Levels of luminance applicable to VMS.

#### Schema Component Representation

```
<xs:simpleType name="VmsLuminanceLevelEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="switchedOff"/>
    <xs:enumeration value="testing"/>
    <xs:enumeration value="night"/>
    <xs:enumeration value="overcast"/>
    <xs:enumeration value="broadDaylight"/>
    <xs:enumeration value="sunInEyes"/>
    <xs:enumeration value="sunOnBack"/>
    <xs:enumeration value="foggyDay"/>
    <xs:enumeration value="foggyNight"/>
  </xs:restriction>
</xs:simpleType>
```

[top](#)

### Simple Type: VmsMessageInformationTypeEnum

**Super-types:** [xs:string](#) < **VmsMessageInformationTypeEnum** (by restriction)

**Sub-types:** None

- Name** VmsMessageInformationTypeEnum
- Content**
- Base XSD Type: string
  - *value* comes from list:  
{campaignMessage|dateTime|futureInformation|instructionOrMessage|situationWarning|temperature|trafficManagement|travelTime}
- Documentation** Types of information displayable on a VMS.

#### Schema Component Representation

```
<xs:simpleType name="VmsMessageInformationTypeEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="campaignMessage"/>
    <xs:enumeration value="dateTime"/>
    <xs:enumeration value="futureInformation"/>
    <xs:enumeration value="instructionOrMessage"/>
    <xs:enumeration value="situationWarning"/>
    <xs:enumeration value="temperature"/>
    <xs:enumeration value="trafficManagement"/>
    <xs:enumeration value="travelTime"/>
  </xs:restriction>
</xs:simpleType>
```

[top](#)