

CEN/TS 16157 Truck Parking Table Publication profile

Table of Contents

- [Schema Document Properties](#)
- [Global Declarations](#)
 - [Element: **d2LogicalModel**](#)
- [Global Definitions](#)
 - [Complex Type: **AcceptedPaymentCards**](#)
 - [Complex Type: **AffectedCarriagewayAndLanes**](#)
 - [Complex Type: **AlertCArea**](#)
 - [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: **Area**](#)
 - [Complex Type: **AreaDestination**](#)
 - [Complex Type: **AreaExtended**](#)
 - [Complex Type: **Charge**](#)
 - [Complex Type: **ChargeBand**](#)
 - [Complex Type: **ChargeBandByReference**](#)
 - [Complex Type: **Contact**](#)
 - [Complex Type: **ContactByReference**](#)
 - [Complex Type: **ContactDetails**](#)
 - [Complex Type: **D2LogicalModel**](#)
 - [Complex Type: **DayWeekMonth**](#)
 - [Complex Type: **DedicatedAccess**](#)
 - [Complex Type: **Destination**](#)
 - [Complex Type: **Dimension**](#)
 - [Complex Type: **DistanceAlongLinearElement**](#)
 - [Complex Type: **DistanceFromLinearElementReferent**](#)
 - [Complex Type: **DistanceFromLinearElementStart**](#)
 - [Complex Type: **ElectricCharging**](#)
 - [Complex Type: **Equipment**](#)
 - [Complex Type: **Exchange**](#)
 - [Complex Type: **ExternalReferencing**](#)
 - [Complex Type: **GenericPublication**](#)
 - [Complex Type: **GrossWeightCharacteristic**](#)
 - [Complex Type: **GroupOfLocations**](#)
 - [Complex Type: **GroupOfParkingSites**](#)
 - [Complex Type: **GroupOfParkingSpaces**](#)
 - [Complex Type: **HazardousMaterials**](#)
 - [Complex Type: **HeaderInformation**](#)
 - [Complex Type: **HeaviestAxleWeightCharacteristic**](#)
 - [Complex Type: **HeightCharacteristic**](#)
 - [Complex Type: **InterUrbanParkingSite**](#)
 - [Complex Type: **InternationalIdentifier**](#)
 - [Complex Type: **Itinerary**](#)
 - [Complex Type: **ItineraryByIndexedLocations**](#)

- o [Complex Type: ItineraryByReference](#)
- o [Complex Type: Junction](#)
- o [Complex Type: LengthCharacteristic](#)
- o [Complex Type: Linear](#)
- o [Complex Type: LinearElement](#)
- o [Complex Type: LinearElementByCode](#)
- o [Complex Type: LinearElementByPoints](#)
- o [Complex Type: LinearWithinLinearElement](#)
- o [Complex Type: Location](#)
- o [Complex Type: LocationByReference](#)
- o [Complex Type: MultilingualString](#)
- o [Complex Type: MultilingualStringValue](#)
- o [Complex Type: NamedArea](#)
- o [Complex Type: NetworkLocation](#)
- o [Complex Type: NonOrderedLocationGroupByList](#)
- o [Complex Type: NonOrderedLocationGroupByReference](#)
- o [Complex Type: NonOrderedLocations](#)
- o [Complex Type: NumberOfAxlesCharacteristic](#)
- o [Complex Type: OffsetDistance](#)
- o [Complex Type: OpeningTimes](#)
- o [Complex Type: OverallPeriod](#)
- o [Complex Type: ParkingAccess](#)
- o [Complex Type: ParkingAssignment](#)
- o [Complex Type: ParkingEquipmentOrServiceFacility](#)
- o [Complex Type: ParkingPermit](#)
- o [Complex Type: ParkingRecord](#)
- o [Complex Type: ParkingRoute](#)
- o [Complex Type: ParkingRouteByReference](#)
- o [Complex Type: ParkingRouteDetails](#)
- o [Complex Type: ParkingSite](#)
- o [Complex Type: ParkingSpace](#)
- o [Complex Type: ParkingSpaceBasics](#)
- o [Complex Type: ParkingStandardsAndSecurity](#)
- o [Complex Type: ParkingStatusColourMapping](#)
- o [Complex Type: ParkingTable](#)
- o [Complex Type: ParkingTablePublication](#)
- o [Complex Type: ParkingThresholds](#)
- o [Complex Type: ParkingUsageScenario](#)
- o [Complex Type: ParkingVMS](#)
- o [Complex Type: PayloadPublication](#)
- o [Complex Type: PercentageDistanceAlongLinearElement](#)
- o [Complex Type: Period](#)
- o [Complex Type: PeriodExtended](#)
- o [Complex Type: PermitsAndProhibitions](#)
- o [Complex Type: Point](#)
- o [Complex Type: PointAlongLinearElement](#)
- o [Complex Type: PointByCoordinates](#)
- o [Complex Type: PointCoordinates](#)
- o [Complex Type: PointDestination](#)
- o [Complex Type: PointExtended](#)
- o [Complex Type: PolygonArea](#)
- o [Complex Type: PublicHoliday](#)
- o [Complex Type: RGBColour](#)
- o [Complex Type: Reference](#)
- o [Complex Type: Referent](#)
- o [Complex Type: Road](#)
- o [Complex Type: RoadNode](#)
- o [Complex Type: ServiceFacility](#)
- o [Complex Type: SpecialDay](#)
- o [Complex Type: SpecialLocationParkingSite](#)
- o [Complex Type: SupplementaryPositionalDescription](#)
- o [Complex Type: TariffsAndPayment](#)
- o [Complex Type: TimePeriodByHour](#)
- o [Complex Type: TimePeriodOfDay](#)

- o [Complex Type: TpegAreaDescriptor](#)
- o [Complex Type: TpegAreaLocation](#)
- o [Complex Type: TpegDescriptor](#)
- o [Complex Type: TpegFramedPoint](#)
- o [Complex Type: TpegGeometricArea](#)
- o [Complex Type: TpegHeight](#)
- o [Complex Type: TpegIlcPointDescriptor](#)
- o [Complex Type: TpegJunction](#)
- o [Complex Type: TpegJunctionPointDescriptor](#)
- o [Complex Type: TpegLinearLocation](#)
- o [Complex Type: TpegNamedOnlyArea](#)
- o [Complex Type: TpegNonJunctionPoint](#)
- o [Complex Type: TpegOtherPointDescriptor](#)
- o [Complex Type: TpegPoint](#)
- o [Complex Type: TpegPointDescriptor](#)
- o [Complex Type: TpegPointLocation](#)
- o [Complex Type: TpegSimplePoint](#)
- o [Complex Type: UrbanParkingSite](#)
- o [Complex Type: Validity](#)
- o [Complex Type: VehicleCharacteristics](#)
- o [Complex Type: VehicleCharacteristicsExtended](#)
- o [Complex Type: VersionedReference](#)
- o [Complex Type: WidthCharacteristic](#)
- o [Complex Type: AreaExtensionType](#)
- o [Complex Type: ChargeBandVersionedReference](#)
- o [Complex Type: ContactDetailsVersionedReference](#)
- o [Complex Type: ExtensionType](#)
- o [Complex Type: GenericPublicationExtensionType](#)
- o [Complex Type: GroupOfParkingSpaces](#)
- o [Complex Type: GroupOfParkingSpacesParkingSpaceIndexParkingSpace](#)
- o [Complex Type: IntermediatePointOnLinearElement](#)
- o [Complex Type: LocationContainedInItinerary](#)
- o [Complex Type: ParkingAccessReference](#)
- o [Complex Type: ParkingRecordEquipmentOrServiceFacilityIndexParkingEquipmentOrServiceFacility](#)
- o [Complex Type: ParkingRecordVersionedReference](#)
- o [Complex Type: ParkingRouteDetailsVersionedReference](#)
- o [Complex Type: ParkingSiteScenarioIndexParkingUsageScenario](#)
- o [Complex Type: ParkingSpace](#)
- o [Complex Type: ParkingSpaceBasicsEquipmentOrServiceFacilityIndexParkingEquipmentOrServiceFacility](#)
- o [Complex Type: ParkingSpaceBasicsScenarioIndexParkingUsageScenario](#)
- o [Complex Type: PeriodExtensionType](#)
- o [Complex Type: PointExtensionType](#)
- o [Complex Type: PolygonAreaIndexPointCoordinates](#)
- o [Complex Type: PredefinedItineraryVersionedReference](#)
- o [Complex Type: PredefinedLocationVersionedReference](#)
- o [Complex Type: PredefinedNonOrderedLocationGroupVersionedReference](#)
- o [Complex Type: VehicleCharacteristicsExtensionType](#)
- o [Complex Type: VmsUnitRecordVersionedReference](#)
- o [Simple Type: AccessCategoryEnum](#)
- o [Simple Type: AccessEquipmentEnum](#)
- o [Simple Type: AccessibilityEnum](#)
- o [Simple Type: AlertCDirectionEnum](#)
- o [Simple Type: AlertCLocationCode](#)
- o [Simple Type: AmountOfMoney](#)
- o [Simple Type: Ampere](#)
- o [Simple Type: AreaOfInterestEnum](#)
- o [Simple Type: AvailabilityEnum](#)
- o [Simple Type: Boolean](#)
- o [Simple Type: CarriagewayEnum](#)
- o [Simple Type: ChargeTypeEnum](#)
- o [Simple Type: ChargingStationUsageTypeEnum](#)
- o [Simple Type: ComparisonOperatorEnum](#)

- o Simple Type: [ConfidentialityValueEnum](#)
- o Simple Type: [CountryEnum](#)
- o Simple Type: [CubicMetres](#)
- o Simple Type: [CurrencyEnum](#)
- o Simple Type: [DangerousGoodsRegulationsEnum](#)
- o Simple Type: [Date](#)
- o Simple Type: [DateTime](#)
- o Simple Type: [DayEnum](#)
- o Simple Type: [Decimal](#)
- o Simple Type: [DirectionEnum](#)
- o Simple Type: [EquipmentTypeEnum](#)
- o Simple Type: [Float](#)
- o Simple Type: [FuelTypeEnum](#)
- o Simple Type: [GroupOfParkingSitesTypeEnum](#)
- o Simple Type: [HeightGradeEnum](#)
- o Simple Type: [IndexReference](#)
- o Simple Type: [InformationStatusEnum](#)
- o Simple Type: [Integer](#)
- o Simple Type: [InterUrbanParkingSiteLocationEnum](#)
- o Simple Type: [JunctionClassificationEnum](#)
- o Simple Type: [LABELSecurityLevelEnum](#)
- o Simple Type: [LABELServiceLevelEnum](#)
- o Simple Type: [LaneEnum](#)
- o Simple Type: [Language](#)
- o Simple Type: [LinearElementNatureEnum](#)
- o Simple Type: [LinearReferencingDirectionEnum](#)
- o Simple Type: [LoadType2Enum](#)
- o Simple Type: [LoadTypeEnum](#)
- o Simple Type: [LocationDescriptorEnum](#)
- o Simple Type: [MeansOfPaymentEnum](#)
- o Simple Type: [MetresAsFloat](#)
- o Simple Type: [MetresAsNonNegativeInteger](#)
- o Simple Type: [MonthOfYearEnum](#)
- o Simple Type: [MultilingualStringValue](#)
- o Simple Type: [NonNegativeInteger](#)
- o Simple Type: [OccupancyDetectionTypeEnum](#)
- o Simple Type: [OwnershipTypeEnum](#)
- o Simple Type: [ParkingDurationEnum](#)
- o Simple Type: [ParkingLayoutEnum](#)
- o Simple Type: [ParkingModeEnum](#)
- o Simple Type: [ParkingPaymentModeEnum](#)
- o Simple Type: [ParkingRouteDirectionEnum](#)
- o Simple Type: [ParkingRouteTypeEnum](#)
- o Simple Type: [ParkingSecurityEnum](#)
- o Simple Type: [ParkingSiteStatusEnum](#)
- o Simple Type: [ParkingSpaceAccessibilityEnum](#)
- o Simple Type: [ParkingSpacePhysicsEnum](#)
- o Simple Type: [ParkingSpecialLocationEnum](#)
- o Simple Type: [ParkingSupervisionEnum](#)
- o Simple Type: [ParkingTypeOfGroup](#)
- o Simple Type: [ParkingUsageScenarioEnum](#)
- o Simple Type: [PaymentCardBrandsEnum](#)
- o Simple Type: [PaymentCardTypesEnum](#)
- o Simple Type: [Percentage](#)
- o Simple Type: [PermitTypeEnum](#)
- o Simple Type: [PublicHolidayTypeEnum](#)
- o Simple Type: [ReferentTypeEnum](#)
- o Simple Type: [RegulationEnum](#)
- o Simple Type: [ReservationTypeEnum](#)
- o Simple Type: [ResponseEnum](#)
- o Simple Type: [RestAreaActivityEnum](#)
- o Simple Type: [RoadTypeEnum](#)
- o Simple Type: [Seconds](#)
- o Simple Type: [ServiceFacilityTypeEnum](#)

- o Simple Type: [SpecialDayTypeEnum](#)
- o Simple Type: [SquareMetres](#)
- o Simple Type: [String](#)
- o Simple Type: [TemperatureCelsius](#)
- o Simple Type: [Time](#)
- o Simple Type: [Tonnes](#)
- o Simple Type: [TpegLoc01AreaLocationSubtypeEnum](#)
- o Simple Type: [TpegLoc01FramedPointLocationSubtypeEnum](#)
- o Simple Type: [TpegLoc01LinearLocationSubtypeEnum](#)
- o Simple Type: [TpegLoc01SimplePointLocationSubtypeEnum](#)
- o Simple Type: [TpegLoc03AreaDescriptorSubtypeEnum](#)
- o Simple Type: [TpegLoc03IlcPointDescriptorSubtypeEnum](#)
- o Simple Type: [TpegLoc03JunctionPointDescriptorSubtypeEnum](#)
- o Simple Type: [TpegLoc03OtherPointDescriptorSubtypeEnum](#)
- o Simple Type: [TpegLoc04HeightTypeEnum](#)
- o Simple Type: [TruckParkingDynamicManagementEnum](#)
- o Simple Type: [UrbanParkingSiteTypeEnum](#)
- o Simple Type: [UrgencyEnum](#)
- o Simple Type: [Url](#)
- o Simple Type: [UserTypeEnum](#)
- o Simple Type: [ValidityStatusEnum](#)
- o Simple Type: [VehicleEquipmentEnum](#)
- o Simple Type: [VehicleType2Enum](#)
- o Simple Type: [VehicleTypeEnum](#)
- o Simple Type: [Volt](#)
- o Simple Type: [WeekOfMonthEnum](#)

[top](#)

Schema Document Properties

Target Namespace http://datex2.eu/schema/2/2_0

Version 2.3

Element and Attribute Namespaces

- 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	http://www.w3.org/XML/1998/namespace
xs	http://www.w3.org/2001/XMLSchema
D2LogicalModel	http://datex2.eu/schema/2/2_0

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**

Name	d2LogicalModel
Type	D2LogicalModel:D2LogicalModel
Nilable	no
Abstract	no

XML Instance Representation

```
<D2LogicalModel:d2LogicalModel
  modelBaseVersion="2 [1]">
  <!--
    Uniqueness Constraint -
    _d2LogicalModelParkingRouteDetailsConstraint
    Selector - ../D2LogicalModel:parkingRouteDetails
    Field(s) - @id, @version
    -->
  <!--
    Uniqueness Constraint - _d2LogicalModelParkingTableConstraint
    Selector - ../D2LogicalModel:parkingTable
    Field(s) - @id, @version
    -->
  <!--
    Uniqueness Constraint - _d2LogicalModelContactDetailsConstraint
    Selector - ../D2LogicalModel:contactDetails
    Field(s) - @id, @version
    -->
  <!--
    Uniqueness Constraint - _d2LogicalModelParkingAccessConstraint
    Selector - ../D2LogicalModel:parkingAccess
    Field(s) - @id
    -->
  <!--
    Uniqueness Constraint - _d2LogicalModelParkingRecordConstraint
    Selector - ../D2LogicalModel:parkingRecord
    Field(s) - @id, @version
    -->
  <!--
    Uniqueness Constraint - _d2LogicalModelChargeBandConstraint
    Selector - ../D2LogicalModel:chargeBand
    Field(s) - @id, @version
    -->

  <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">
  <xs:unique name="_d2LogicalModelParkingRouteDetailsConstraint">
    <xs:selector xpath="../D2LogicalModel:parkingRouteDetails"/>
```

```

        <xs:field xpath="@id"/>
        <xs:field xpath="@version"/>
    </xs:unique>
    <xs:unique name="_d2LogicalModelParkingTableConstraint">
        <xs:selector xpath="//D2LogicalModel:parkingTable"/>
        <xs:field xpath="@id"/>
        <xs:field xpath="@version"/>
    </xs:unique>
    <xs:unique name="_d2LogicalModelContactDetailsConstraint">
        <xs:selector xpath="//D2LogicalModel:contactDetails"/>
        <xs:field xpath="@id"/>
        <xs:field xpath="@version"/>
    </xs:unique>
    <xs:unique name="_d2LogicalModelParkingAccessConstraint">
        <xs:selector xpath="//D2LogicalModel:parkingAccess"/>
        <xs:field xpath="@id"/>
    </xs:unique>
    <xs:unique name="_d2LogicalModelParkingRecordConstraint">
        <xs:selector xpath="//D2LogicalModel:parkingRecord"/>
        <xs:field xpath="@id"/>
        <xs:field xpath="@version"/>
    </xs:unique>
    <xs:unique name="_d2LogicalModelChargeBandConstraint">
        <xs:selector xpath="//D2LogicalModel:chargeBand"/>
        <xs:field xpath="@id"/>
        <xs:field xpath="@version"/>
    </xs:unique>
</xs:element>

```

[top](#)

Global Definitions

Complex Type: **AcceptedPaymentCards**

Super-types: None

Sub-types: None

Name	AcceptedPaymentCards
<u>Abstract</u>	no
Documentation	Use this class to describe details in case acceptedMeansOfPayment is set to 'paymentCard'.

XML Instance Representation

```

<...>
  <D2LogicalModel:paymentCards> D2LogicalModel:PaymentCardTypeEnum
</D2LogicalModel:paymentCards> [1..*] ?
  <D2LogicalModel:otherPaymentCards> D2LogicalModel:String
</D2LogicalModel:otherPaymentCards> [0..*] ?
  <D2LogicalModel:paymentCardBrands>
D2LogicalModel:PaymentCardBrandsEnum
</D2LogicalModel:paymentCardBrands> [0..*] ?
  <D2LogicalModel:otherPaymentCardBrands> D2LogicalModel:String
</D2LogicalModel:otherPaymentCardBrands> [0..*] ?
  <D2LogicalModel:acceptedPaymentCardsExtension>
D2LogicalModel:ExtensionType
</D2LogicalModel:acceptedPaymentCardsExtension> [0..1]
</...>

```

Schema Component Representation

```

<xs:complexType name="AcceptedPaymentCards">
  <xs:sequence>
    <xs:element name="paymentCards"
      type="D2LogicalModel:PaymentCardTypeEnum" minOccurs="1"
      maxOccurs="unbounded"/>
    <xs:element name="otherPaymentCards"
      type="D2LogicalModel:String" minOccurs="0"
      maxOccurs="unbounded"/>
    <xs:element name="paymentCardBrands"
      type="D2LogicalModel:PaymentCardBrandsEnum" minOccurs="0"
      maxOccurs="unbounded"/>
    <xs:element name="otherPaymentCardBrands"
      type="D2LogicalModel:String" minOccurs="0"
      maxOccurs="unbounded"/>
    <xs:element name="acceptedPaymentCardsExtension"
      type="D2LogicalModel:ExtensionType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>

```

[top](#)

Complex Type: **AffectedCarriagewayAndLanes**

Super-types: None

Sub-types: None

Name AffectedCarriagewayAndLanes

Abstract no

Documentation 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: AlertCArea

Super-types: None

Sub-types: None

Name	AlertCArea
Abstract	no
Documentation	An area defined by reference to a predefined 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:areaLocation> D2LogicalModel:AlertCLocation
  </D2LogicalModel:areaLocation> [1] ?
  <D2LogicalModel:alertCAreaExtension> D2LogicalModel: _ExtensionType
  </D2LogicalModel:alertCAreaExtension> [0..1]
</...>

```

Schema Component Representation

```

<xs:complexType name="AlertCArea">
  <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="areaLocation"
  type="D2LogicalModel:AlertCLocation"/>
<xs:element name="alertCAreaExtension"
  type="D2LogicalModel:ExtensionType" minOccurs="0"/>
</xs:sequence>
</xs:complexType>

```

[top](#)

Complex Type: **AlertCDirection**

Super-types:	None
Sub-types:	None

Name	AlertCDirection
Abstract	no
Documentation	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:

- [AlertCLinearByCode](#) (by extension)
- [AlertCMethod2Linear](#) (by extension)
- [AlertCMethod4Linear](#) (by extension)

Name AlertCLinear

Abstract yes

Documentation 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: [AlertCLinear](#) < AlertCLinearByCode (by extension)

Sub-types: None

Name AlertCLinearByCode

Abstract no

Documentation 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

Name AlertCLocation

Abstract no

Documentation 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: [AlertCLinear](#) < **AlertCMethod2Linear** (by extension)

Sub-types: None

Name AlertCMethod2Linear

Abstract no

Documentation 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: [AlertCPoint](#) < AlertCMethod2Point (by extension)

Sub-types: None

Name	AlertCMethod2Point
Abstract	no
Documentation	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

Name AlertCMethod2PrimaryPointLocation

Abstract no

Documentation 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

Name AlertCMethod2SecondaryPointLocation

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.

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: [AlertCLinear](#) < AlertCMethod4Linear (by extension)

Sub-types: None

Name AlertCMethod4Linear

Abstract no

Documentation 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: [AlertCPoint](#) < AlertCMethod4Point (by extension)

Sub-types: None

Name AlertCMethod4Point

Abstract no

Documentation 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

Name AlertCMethod4PrimaryPointLocation

Abstract no

Documentation 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: **Area**

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

Sub-types: None

Name	Area
Abstract	no
Documentation	A geographic or geometric defined area which may be qualified by height information to provide additional geospatial discrimination (e.g. for snow in an area but only above a certain altitude).

XML Instance Representation

```

<...>
  <D2LogicalModel:groupOfLocationsExtension>
    D2LogicalModel:_ExtensionType
  </D2LogicalModel:groupOfLocationsExtension> [0..1]
  <D2LogicalModel:externalReferencing>
    D2LogicalModel:ExternalReferencing
  </D2LogicalModel:externalReferencing> [0..*]
  <D2LogicalModel:locationForDisplay> D2LogicalModel:PointCoordinates
  </D2LogicalModel:locationForDisplay> [0..1] ?
  <D2LogicalModel:locationExtension> D2LogicalModel:_ExtensionType
  </D2LogicalModel:locationExtension> [0..1]
  <D2LogicalModel:alertCArea> D2LogicalModel:AlertCArea
  </D2LogicalModel:alertCArea> [0..1]
  <D2LogicalModel:tpegAreaLocation> D2LogicalModel:TpegAreaLocation
  </D2LogicalModel:tpegAreaLocation> [0..1]
  <D2LogicalModel:areaExtension> D2LogicalModel:_AreaExtensionType
  </D2LogicalModel:areaExtension> [0..1]
</...>

```

Schema Component Representation

```

<xs:complexType name="Area">
  <xs:complexContent>
    <xs:extension base="D2LogicalModel:Location">
      <xs:sequence>
        <xs:element name="alertCArea"
          type="D2LogicalModel:AlertCArea" minOccurs="0"/>
        <xs:element name="tpegAreaLocation"
          type="D2LogicalModel:TpegAreaLocation" minOccurs="0"/>
        <xs:element name="areaExtension"
          type="D2LogicalModel:AreaExtensionType" minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

```

[top](#)

Complex Type: **AreaDestination**

Super-types: [Destination](#) < **AreaDestination** (by extension)

Sub-types: None

Name AreaDestination

Abstract no

Documentation The specification of the destination of a defined route or itinerary which is an area.

XML Instance Representation

```

<...>
  <D2LogicalModel:destinationExtension> D2LogicalModel:_ExtensionType
</D2LogicalModel:destinationExtension> [0..1]
  <D2LogicalModel:area> D2LogicalModel:Area </D2LogicalModel:area>
  [1]
  <D2LogicalModel:areaDestinationExtension>
    D2LogicalModel:_ExtensionType
  </D2LogicalModel:areaDestinationExtension> [0..1]
</...>

```

Schema Component Representation

```

<xs:complexType name="AreaDestination">
  <xs:complexContent>
    <xs:extension base="D2LogicalModel:Destination">
      <xs:sequence>
        <xs:element name="area" type="D2LogicalModel:Area"/>
        <xs:element name="areaDestinationExtension"
          type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

```

[top](#)

Complex Type: **AreaExtended**

Super-types: None

Sub-types: None

Name AreaExtended

Abstract no

Documentation Extension class for area used in parking publication extension.

XML Instance Representation

```
<...>
  <D2LogicalModel:namedArea> D2LogicalModel:NamedArea
</D2LogicalModel:namedArea> [0..1]
  <D2LogicalModel:polygonArea> D2LogicalModel:PolygonArea
</D2LogicalModel:polygonArea> [0..*]
</...>
```

Schema Component Representation

```
<xs:complexType name="AreaExtended">
  <xs:sequence>
    <xs:element name="namedArea" type="D2LogicalModel:NamedArea"
      minOccurs="0"/>
    <xs:element name="polygonArea" type="D2LogicalModel:PolygonArea"
      minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>
```

[top](#)

Complex Type: Charge

Super-types: None

Sub-types: None

Name Charge

Abstract no

Documentation A particular charge for a specified interval belonging a charge band.

XML Instance Representation

```
<...>
  <D2LogicalModel:charge> D2LogicalModel:AmountOfMoney
</D2LogicalModel:charge> [1] ?
  <D2LogicalModel:chargeInterval> D2LogicalModel:Seconds
</D2LogicalModel:chargeInterval> [0..1] ?
  <D2LogicalModel:chargeType> D2LogicalModel:ChargeTypeEnum
</D2LogicalModel:chargeType> [0..1] ?
  <D2LogicalModel:chargeTypeDescription>
    D2LogicalModel:MultilingualString
  </D2LogicalModel:chargeTypeDescription> [0..1] ?
  <D2LogicalModel:maxIterationsOfCharge>
    D2LogicalModel:NonNegativeInteger
  </D2LogicalModel:maxIterationsOfCharge> [0..1] ?
  <D2LogicalModel:minIterationsOfCharge>
    D2LogicalModel:NonNegativeInteger
```

```

</D2LogicalModel:minIterationsOfCharge> [0..1] ?
<D2LogicalModel:chargeOrderIndex> D2LogicalModel:NonNegativeInteger
</D2LogicalModel:chargeOrderIndex> [0..1] ?
<D2LogicalModel:timePeriodOfDay> D2LogicalModel:TimePeriodOfDay
</D2LogicalModel:timePeriodOfDay> [0..1] ?
<D2LogicalModel:chargeExtension> D2LogicalModel:_ExtensionType
</D2LogicalModel:chargeExtension> [0..1]
</...>

```

Schema Component Representation

```

<xs:complexType name="Charge">
  <xs:sequence>
    <xs:element name="charge" type="D2LogicalModel:AmountOfMoney"
      minOccurs="1" maxOccurs="1"/>
    <xs:element name="chargeInterval" type="D2LogicalModel:Seconds"
      minOccurs="0" maxOccurs="1"/>
    <xs:element name="chargeType"
      type="D2LogicalModel:ChargeTypeEnum" minOccurs="0"
      maxOccurs="1"/>
    <xs:element name="chargeTypeDescription"
      type="D2LogicalModel:MultilingualString" minOccurs="0"
      maxOccurs="1"/>
    <xs:element name="maxIterationsOfCharge"
      type="D2LogicalModel:NonNegativeInteger" minOccurs="0"
      maxOccurs="1"/>
    <xs:element name="minIterationsOfCharge"
      type="D2LogicalModel:NonNegativeInteger" minOccurs="0"
      maxOccurs="1"/>
    <xs:element name="chargeOrderIndex"
      type="D2LogicalModel:NonNegativeInteger" minOccurs="0"
      maxOccurs="1"/>
    <xs:element name="timePeriodOfDay"
      type="D2LogicalModel:TimePeriodOfDay" minOccurs="0"/>
    <xs:element name="chargeExtension"
      type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>

```

[top](#)

Complex Type: **ChargeBand**

Super-types: None

Sub-types: None

Name ChargeBand

Abstract no

Documentation A charge band in accordance with the specified conditions, possibly up to a maximum duration, during a specified period and for a vehicle of specified characteristics (in case of parking).

XML Instance Representation

```

<...
  id="xs:string [1]"
  version="xs:string [1]">
    <D2LogicalModel:chargeCurrency> D2LogicalModel:CurrencyEnum

```

```

</D2LogicalModel:chargeCurrency> [1] ?
<D2LogicalModel:maximumDuration> D2LogicalModel:Seconds
</D2LogicalModel:maximumDuration> [0..1] ?
<D2LogicalModel:chargeBandName> D2LogicalModel:MultilingualString
</D2LogicalModel:chargeBandName> [0..1] ?
<D2LogicalModel:applicableForUser> D2LogicalModel:UserTypeEnum
</D2LogicalModel:applicableForUser> [0..*] ?
<D2LogicalModel:charge> D2LogicalModel:Charge
</D2LogicalModel:charge> [1..*]
<D2LogicalModel:applicableForPeriod> D2LogicalModel:OverallPeriod
</D2LogicalModel:applicableForPeriod> [0..1] ?
<D2LogicalModel:applicableForVehicles>
D2LogicalModel:VehicleCharacteristics
</D2LogicalModel:applicableForVehicles> [0..*] ?
<D2LogicalModel:parkingPermit> D2LogicalModel:ParkingPermit
</D2LogicalModel:parkingPermit> [0..*]
<D2LogicalModel:chargeBandExtension> D2LogicalModel:_ExtensionType
</D2LogicalModel:chargeBandExtension> [0..1]
</...>

```

Schema Component Representation

```

<xs:complexType name="ChargeBand">
  <xs:sequence>
    <xs:element name="chargeCurrency"
      type="D2LogicalModel:CurrencyEnum" minOccurs="1" maxOccurs="1"/>
    <xs:element name="maximumDuration" type="D2LogicalModel:Seconds"
      minOccurs="0" maxOccurs="1"/>
    <xs:element name="chargeBandName"
      type="D2LogicalModel:MultilingualString" minOccurs="0"
      maxOccurs="1"/>
    <xs:element name="applicableForUser"
      type="D2LogicalModel:UserTypeEnum" minOccurs="0"
      maxOccurs="unbounded"/>
    <xs:element name="charge" type="D2LogicalModel:Charge"
      maxOccurs="unbounded"/>
    <xs:element name="applicableForPeriod"
      type="D2LogicalModel:OverallPeriod" minOccurs="0"/>
    <xs:element name="applicableForVehicles"
      type="D2LogicalModel:VehicleCharacteristics" minOccurs="0"
      maxOccurs="unbounded"/>
    <xs:element name="parkingPermit"
      type="D2LogicalModel:ParkingPermit" minOccurs="0"
      maxOccurs="unbounded"/>
    <xs:element name="chargeBandExtension"
      type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
  </xs:sequence>
  <xs:attribute name="id" type="xs:string" use="required"/>
  <xs:attribute name="version" type="xs:string" use="required"/>
</xs:complexType>

```

[top](#)

Complex Type: ChargeBandByReference

Super-types:	None
Sub-types:	None

Name	ChargeBandByReference
------	-----------------------

Abstract	no
Documentation	Using (a) prior defined charge band(s), identified by its reference.

XML Instance Representation

```
<...>
  <D2LogicalModel:chargeBandReference>
    D2LogicalModel: _ChargeBandVersionedReference
  </D2LogicalModel:chargeBandReference> [1] ?
  <D2LogicalModel:chargeBandByReferenceExtension>
    D2LogicalModel: _ExtensionType
  </D2LogicalModel:chargeBandByReferenceExtension> [0..1]
</...>
```

Schema Component Representation

```
<xs:complexType name="ChargeBandByReference">
  <xs:sequence>
    <xs:element name="chargeBandReference"
      type="D2LogicalModel: _ChargeBandVersionedReference"
      minOccurs="1" maxOccurs="1"/>
    <xs:element name="chargeBandByReferenceExtension"
      type="D2LogicalModel: _ExtensionType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
```

[top](#)

Complex Type: Contact

Super-types:	None
Sub-types:	ContactByReference (by extension) ContactDetails (by extension)

Name	Contact
Abstract	no
Documentation	Address and contact information about some person, service or the parking site, provided in detail or via reference.

XML Instance Representation

```
<...>
  <D2LogicalModel:contactUnknown> D2LogicalModel:Boolean
</D2LogicalModel:contactUnknown> [0..1] ?
  <D2LogicalModel:contactNotDefined> D2LogicalModel:Boolean
</D2LogicalModel:contactNotDefined> [0..1] ?
  <D2LogicalModel:validityOfContact> D2LogicalModel:OverallPeriod
</D2LogicalModel:validityOfContact> [0..1]
  <D2LogicalModel:contactExtension> D2LogicalModel: _ExtensionType
</D2LogicalModel:contactExtension> [0..1]
</...>
```

Schema Component Representation

```
<xs:complexType name="Contact">
  <xs:sequence>
    <xs:element name="contactUnknown" type="D2LogicalModel:Boolean"
```

```

        minOccurs="0" maxOccurs="1"/>
        <xs:element name="contactNotDefined"
        type="D2LogicalModel:Boolean" minOccurs="0" maxOccurs="1"/>
        <xs:element name="validityOfContact"
        type="D2LogicalModel:OverallPeriod" minOccurs="0"/>
        <xs:element name="contactExtension"
        type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
    </xs:sequence>
</xs:complexType>

```

[top](#)

Complex Type: **ContactByReference**

Super-types: [Contact](#) < **ContactByReference** (by extension)

Sub-types: None

Name	ContactByReference
Abstract	no
Documentation	Contact information that is addressed via a reference.

XML Instance Representation

```

<...>
  <D2LogicalModel:contactUnknown> D2LogicalModel:Boolean
</D2LogicalModel:contactUnknown> [0..1] ?
  <D2LogicalModel:contactNotDefined> D2LogicalModel:Boolean
</D2LogicalModel:contactNotDefined> [0..1] ?
  <D2LogicalModel:validityOfContact> D2LogicalModel:OverallPeriod
</D2LogicalModel:validityOfContact> [0..1]
  <D2LogicalModel:contactExtension> D2LogicalModel:_ExtensionType
</D2LogicalModel:contactExtension> [0..1]
  <D2LogicalModel:contactReference>
    D2LogicalModel:_ContactDetailsVersionedReference
  </D2LogicalModel:contactReference> [1] ?
  <D2LogicalModel:contactByReferenceExtension>
    D2LogicalModel:_ExtensionType
  </D2LogicalModel:contactByReferenceExtension> [0..1]
</...>

```

Schema Component Representation

```

<xs:complexType name="ContactByReference">
  <xs:complexContent>
    <xs:extension base="D2LogicalModel:Contact">
      <xs:sequence>
        <xs:element name="contactReference"
          type="D2LogicalModel:_ContactDetailsVersionedReference"
          minOccurs="1" maxOccurs="1"/>
        <xs:element name="contactByReferenceExtension"
          type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

```

[top](#)

Complex Type: **ContactDetails**

Super-types: [Contact](#) < **ContactDetails** (by extension)

Sub-types: None

Name	ContactDetails
Abstract	no
Documentation	Details for some person, service or the parking site itself, especially address information.

XML Instance Representation

```
<...  
  id="xs:string [1]"  
  version="xs:string [1]">  
    <D2LogicalModel:contactUnknown> D2LogicalModel:Boolean  
    </D2LogicalModel:contactUnknown> [0..1] ?  
    <D2LogicalModel:contactNotDefined> D2LogicalModel:Boolean  
    </D2LogicalModel:contactNotDefined> [0..1] ?  
    <D2LogicalModel:validityOfContact> D2LogicalModel:OverallPeriod  
    </D2LogicalModel:validityOfContact> [0..1]  
    <D2LogicalModel:contactExtension> D2LogicalModel:_ExtensionType  
    </D2LogicalModel:contactExtension> [0..1]  
    <D2LogicalModel:contactOrganisationName>  
    D2LogicalModel:MultilingualString  
    </D2LogicalModel:contactOrganisationName> [0..1] ?  
    <D2LogicalModel:contactPersonName> D2LogicalModel:String  
    </D2LogicalModel:contactPersonName> [0..1] ?  
    <D2LogicalModel:contactPersonFirstName> D2LogicalModel:String  
    </D2LogicalModel:contactPersonFirstName> [0..1] ?  
    <D2LogicalModel:contactPersonPosition>  
    D2LogicalModel:MultilingualString  
    </D2LogicalModel:contactPersonPosition> [0..1] ?  
    <D2LogicalModel:contactDetailsLanguage> D2LogicalModel:Language  
    </D2LogicalModel:contactDetailsLanguage> [0..*] ?  
    <D2LogicalModel:contactDetailsAddress>  
    D2LogicalModel:MultilingualString  
    </D2LogicalModel:contactDetailsAddress> [0..1] ?  
    <D2LogicalModel:contactDetailsStreet> D2LogicalModel:String  
    </D2LogicalModel:contactDetailsStreet> [0..1] ?  
    <D2LogicalModel:contactDetailsHouseNumber> D2LogicalModel:String  
    </D2LogicalModel:contactDetailsHouseNumber> [0..2] ?  
    <D2LogicalModel:contactDetailsPostcode> D2LogicalModel:String  
    </D2LogicalModel:contactDetailsPostcode> [0..1] ?  
    <D2LogicalModel:contactDetailsCity>  
    D2LogicalModel:MultilingualString  
    </D2LogicalModel:contactDetailsCity> [0..1] ?  
    <D2LogicalModel:country> D2LogicalModel:CountryEnum  
    </D2LogicalModel:country> [0..1] ?  
    <D2LogicalModel:contactDetailsTelephoneNumber>  
    D2LogicalModel:String  
    </D2LogicalModel:contactDetailsTelephoneNumber> [0..1] ?  
    <D2LogicalModel:contactDetailsFax> D2LogicalModel:String  
    </D2LogicalModel:contactDetailsFax> [0..1] ?  
    <D2LogicalModel:contactDetailsEMail> D2LogicalModel:String  
    </D2LogicalModel:contactDetailsEMail> [0..1] ?  
    <D2LogicalModel:urlLinkAddress> D2LogicalModel:Url  
    </D2LogicalModel:urlLinkAddress> [0..1] ?  
    <D2LogicalModel:contactDetailsLogoUrl> D2LogicalModel:Url
```

```

</D2LogicalModel:contactDetailsLogoUrl> [0..1] ?
<D2LogicalModel:available24hours> D2LogicalModel:Boolean
</D2LogicalModel:available24hours> [0..1] ?
<D2LogicalModel:contactDetailsResponsibility>
D2LogicalModel:MultilingualString
</D2LogicalModel:contactDetailsResponsibility> [0..*] ?
<D2LogicalModel:contactDetailsMoreInfo>
D2LogicalModel:MultilingualString
</D2LogicalModel:contactDetailsMoreInfo> [0..*] ?
<D2LogicalModel:publishingAgreement> D2LogicalModel:Boolean
</D2LogicalModel:publishingAgreement> [0..1] ?
<D2LogicalModel:contactDetailsOwnership>
D2LogicalModel:OwnershipTypeEnum
</D2LogicalModel:contactDetailsOwnership> [0..1] ?
<D2LogicalModel:groupOfLocations> D2LogicalModel:GroupOfLocations
</D2LogicalModel:groupOfLocations> [0..1]
<D2LogicalModel:contactDetailsExtension>
D2LogicalModel:ExtensionType
</D2LogicalModel:contactDetailsExtension> [0..1]
</...>

```

Schema Component Representation

```

<xs:complexType name="ContactDetails">
  <xs:complexContent>
    <xs:extension base="D2LogicalModel:Contact">
      <xs:sequence>
        <xs:element name="contactOrganisationName"
          type="D2LogicalModel:MultilingualString" minOccurs="0"
          maxOccurs="1"/>
        <xs:element name="contactPersonName"
          type="D2LogicalModel:String" minOccurs="0" maxOccurs="1"/>
        <xs:element name="contactPersonFirstName"
          type="D2LogicalModel:String" minOccurs="0" maxOccurs="1"/>
        <xs:element name="contactPersonPosition"
          type="D2LogicalModel:MultilingualString" minOccurs="0"
          maxOccurs="1"/>
        <xs:element name="contactDetailsLanguage"
          type="D2LogicalModel:Language" minOccurs="0"
          maxOccurs="unbounded"/>
        <xs:element name="contactDetailsAddress"
          type="D2LogicalModel:MultilingualString" minOccurs="0"
          maxOccurs="1"/>
        <xs:element name="contactDetailsStreet"
          type="D2LogicalModel:String" minOccurs="0" maxOccurs="1"/>
        <xs:element name="contactDetailsHouseNumber"
          type="D2LogicalModel:String" minOccurs="0" maxOccurs="2"/>
        <xs:element name="contactDetailsPostcode"
          type="D2LogicalModel:String" minOccurs="0" maxOccurs="1"/>
        <xs:element name="contactDetailsCity"
          type="D2LogicalModel:MultilingualString" minOccurs="0"
          maxOccurs="1"/>
        <xs:element name="country"
          type="D2LogicalModel:CountryEnum" minOccurs="0"
          maxOccurs="1"/>
        <xs:element name="contactDetailsTelephoneNumber"
          type="D2LogicalModel:String" minOccurs="0" maxOccurs="1"/>
        <xs:element name="contactDetailsFax"
          type="D2LogicalModel:String" minOccurs="0" maxOccurs="1"/>
        <xs:element name="contactDetailsEMail"
          type="D2LogicalModel:String" minOccurs="0" maxOccurs="1"/>
        <xs:element name="urlLinkAddress" type="D2LogicalModel:Url"
          minOccurs="0" maxOccurs="1"/>
        <xs:element name="contactDetailsLogoUrl"

```

```

type="D2LogicalModel:Url" minOccurs="0" maxOccurs="1"/>
<xs:element name="available24hours"
type="D2LogicalModel:Boolean" minOccurs="0" maxOccurs="1"/>
<xs:element name="contactDetailsResponsibility"
type="D2LogicalModel:MultilingualString" minOccurs="0"
maxOccurs="unbounded"/>
<xs:element name="contactDetailsMoreInfo"
type="D2LogicalModel:MultilingualString" minOccurs="0"
maxOccurs="unbounded"/>
<xs:element name="publishingAgreement"
type="D2LogicalModel:Boolean" minOccurs="0" maxOccurs="1"/>
<xs:element name="contactDetailsOwnership"
type="D2LogicalModel:OwnershipTypeEnum" minOccurs="0"
maxOccurs="1"/>
<xs:element name="groupOfLocations"
type="D2LogicalModel:GroupOfLocations" minOccurs="0"/>
<xs:element name="contactDetailsExtension"
type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
</xs:sequence>
<xs:attribute name="id" type="xs:string" use="required"/>
<xs:attribute name="version" type="xs:string" use="required"/>
</xs:extension>
</xs:complexContent>
</xs:complexType>

```

[top](#)

Complex Type: **D2LogicalModel**

Super-types: None

Sub-types: None

Name	D2LogicalModel
<u>Abstract</u>	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: **DayWeekMonth**

Super-types: None

Sub-types: None

Name DayWeekMonth

Abstract no

Documentation Specification of periods defined by the intersection of days, weeks and months.

XML Instance Representation

```
<...>
  <D2LogicalModel:applicableDay> D2LogicalModel:DayEnum
</D2LogicalModel:applicableDay> [0..7] ?
  <D2LogicalModel:applicableWeek> D2LogicalModel:WeekOfMonthEnum
</D2LogicalModel:applicableWeek> [0..5] ?
  <D2LogicalModel:applicableMonth> D2LogicalModel:MonthOfYearEnum
</D2LogicalModel:applicableMonth> [0..12] ?
  <D2LogicalModel:dayWeekMonthExtension>
D2LogicalModel:ExtensionType
</D2LogicalModel:dayWeekMonthExtension> [0..1]
</...>
```

Schema Component Representation

```
<xs:complexType name="DayWeekMonth">
  <xs:sequence>
    <xs:element name="applicableDay" type="D2LogicalModel:DayEnum"
      minOccurs="0" maxOccurs="7"/>
    <xs:element name="applicableWeek"
      type="D2LogicalModel:WeekOfMonthEnum" minOccurs="0"
      maxOccurs="5"/>
    <xs:element name="applicableMonth"
      type="D2LogicalModel:MonthOfYearEnum" minOccurs="0"
      maxOccurs="12"/>
    <xs:element name="dayWeekMonthExtension"
      type="D2LogicalModel:ExtensionType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
```

[top](#)

Complex Type: **DedicatedAccess**

Super-types: None

Sub-types: None

Name DedicatedAccess

Abstract	no
Documentation	Reference to an access of any type (vehicles, pedestrian, ...).

XML Instance Representation

```
<...>
  <D2LogicalModel:dedicatedAccess>
    D2LogicalModel:_ParkingAccessReference
  </D2LogicalModel:dedicatedAccess> [1] ?
  <D2LogicalModel:distanceFromParkingSpace>
    D2LogicalModel:MetresAsNonNegativeInteger
  </D2LogicalModel:distanceFromParkingSpace> [0..1] ?
  <D2LogicalModel:dedicatedAccessExtension>
    D2LogicalModel:_ExtensionType
  </D2LogicalModel:dedicatedAccessExtension> [0..1]
</...>
```

Schema Component Representation

```
<xs:complexType name="DedicatedAccess">
  <xs:sequence>
    <xs:element name="dedicatedAccess"
      type="D2LogicalModel:_ParkingAccessReference" minOccurs="1"
      maxOccurs="1"/>
    <xs:element name="distanceFromParkingSpace"
      type="D2LogicalModel:MetresAsNonNegativeInteger" minOccurs="0"
      maxOccurs="1"/>
    <xs:element name="dedicatedAccessExtension"
      type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
```

[top](#)

Complex Type: **Destination**

<i>Super-types:</i>	None
<i>Sub-types:</i>	• AreaDestination (by extension) • PointDestination (by extension)

Name	Destination
Abstract	yes
Documentation	The specification a destination. This may be either a point location or an area location.

XML Instance Representation

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

Schema Component Representation

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

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

[top](#)

Complex Type: **Dimension**

Super-types: None

Sub-types: None

Name Dimension

Abstract no

Documentation A component that provides dimension information. The product of width and height must not be necessarily be the square footage (e.g. in multi-storey buildings or when some zones are not part of the square footage).

XML Instance Representation

```
<...>
  <D2LogicalModel:dimensionLength> D2LogicalModel:MetresAsFloat
</D2LogicalModel:dimensionLength> [0..1] ?
  <D2LogicalModel:dimensionWidth> D2LogicalModel:MetresAsFloat
</D2LogicalModel:dimensionWidth> [0..1] ?
  <D2LogicalModel:dimensionHeight> D2LogicalModel:MetresAsFloat
</D2LogicalModel:dimensionHeight> [0..1] ?
  <D2LogicalModel:dimensionUsableArea> D2LogicalModel:SquareMetres
</D2LogicalModel:dimensionUsableArea> [0..1] ?
  <D2LogicalModel:dimensionExtension> D2LogicalModel: _ExtensionType
</D2LogicalModel:dimensionExtension> [0..1]
</...>
```

Schema Component Representation

```
<xs:complexType name="Dimension">
  <xs:sequence>
    <xs:element name="dimensionLength"
      type="D2LogicalModel:MetresAsFloat" minOccurs="0"
      maxOccurs="1"/>
    <xs:element name="dimensionWidth"
      type="D2LogicalModel:MetresAsFloat" minOccurs="0"
      maxOccurs="1"/>
    <xs:element name="dimensionHeight"
      type="D2LogicalModel:MetresAsFloat" minOccurs="0"
      maxOccurs="1"/>
    <xs:element name="dimensionUsableArea"
      type="D2LogicalModel:SquareMetres" minOccurs="0" maxOccurs="1"/>
    <xs:element name="dimensionExtension"
      type="D2LogicalModel: _ExtensionType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
```

[top](#)

Complex Type: **DistanceAlongLinearElement**

Super-types: None

Sub-types:

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

Name	DistanceAlongLinearElement
<u>Abstract</u>	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
<u>Abstract</u>	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>

```

[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: **ElectricCharging**

Super-types: None

Sub-types: None

Name ElectricCharging

Abstract no

Documentation Additional information for the equipment 'electricChargingStation'. This component refers to the number of charging stations specified in the attribute 'numberOfEquipmentOrServiceFacilities'.

XML Instance Representation

```

<...>
  <D2LogicalModel:chargingStationUsageType>
    D2LogicalModel:ChargingStationUsageTypeEnum
  </D2LogicalModel:chargingStationUsageType> [1..*] ?
  <D2LogicalModel:chargingStationModelType>
    D2LogicalModel:MultilingualString
  </D2LogicalModel:chargingStationModelType> [0..1] ?
  <D2LogicalModel:maximumCurrent> D2LogicalModel:Ampere
  </D2LogicalModel:maximumCurrent> [0..1] ?
  <D2LogicalModel:voltage> D2LogicalModel:Volt
  </D2LogicalModel:voltage> [0..*] ?
  <D2LogicalModel:chargingStationConnectorType>
    D2LogicalModel:MultilingualString
  </D2LogicalModel:chargingStationConnectorType> [0..*] ?
  <D2LogicalModel:numberOfChargingPoints>
    D2LogicalModel:NonNegativeInteger
  </D2LogicalModel:numberOfChargingPoints> [0..1] ?
  <D2LogicalModel:electricChargingExtension>
    D2LogicalModel:ExtensionType
  </D2LogicalModel:electricChargingExtension> [0..1]
</...>

```

Schema Component Representation

```

<xs:complexType name="ElectricCharging">
  <xs:sequence>
    <xs:element name="chargingStationUsageType"
      type="D2LogicalModel:ChargingStationUsageTypeEnum" minOccurs="1"
      maxOccurs="unbounded"/>
    <xs:element name="chargingStationModelType"
      type="D2LogicalModel:MultilingualString" minOccurs="0"
      maxOccurs="1"/>
    <xs:element name="maximumCurrent" type="D2LogicalModel:Ampere"
      minOccurs="0" maxOccurs="1"/>
    <xs:element name="voltage" type="D2LogicalModel:Volt"

```

```

minOccurs="0" maxOccurs="unbounded"/>
<xs:element name="chargingStationConnectorType"
type="D2LogicalModel:MultilingualString" minOccurs="0"
maxOccurs="unbounded"/>
<xs:element name="numberOfChargingPoints"
type="D2LogicalModel:NonNegativeInteger" minOccurs="0"
maxOccurs="1"/>
<xs:element name="electricChargingExtension"
type="D2LogicalModel:ExtensionType" minOccurs="0"/>
</xs:sequence>
</xs:complexType>

```

[top](#)

Complex Type: **Equipment**

Super-types: [ParkingEquipmentOrServiceFacility](#) < **Equipment** (by extension)
Sub-types: None

Name	Equipment
<u>Abstract</u>	no
Documentation	One type of equipment, that is available on the parking site.

XML Instance Representation

```

<...>
  <D2LogicalModel:equipmentOrServiceFacilityIdentifier>
    D2LogicalModel:String
  </D2LogicalModel:equipmentOrServiceFacilityIdentifier> [0..*] ?
  <D2LogicalModel:availability> D2LogicalModel:AvailabilityEnum
  </D2LogicalModel:availability> [0..1] ?
  <D2LogicalModel:numberOfEquipmentOrServiceFacility>
    D2LogicalModel:NonNegativeInteger
  </D2LogicalModel:numberOfEquipmentOrServiceFacility> [0..1] ?
  <D2LogicalModel:additionalDescription>
    D2LogicalModel:MultilingualString
  </D2LogicalModel:additionalDescription> [0..1] ?
  <D2LogicalModel:otherEquipmentOrServiceFacility>
    D2LogicalModel:MultilingualString
  </D2LogicalModel:otherEquipmentOrServiceFacility> [0..1] ?
  <D2LogicalModel:accessibility> D2LogicalModel:AccessibilityEnum
  </D2LogicalModel:accessibility> [0..*] ?
  <D2LogicalModel:nameOrBrand> D2LogicalModel:MultilingualString
  </D2LogicalModel:nameOrBrand> [0..1] ?
  <D2LogicalModel:comment> D2LogicalModel:MultilingualString
  </D2LogicalModel:comment> [0..1] ?
  <D2LogicalModel:photoUrl> D2LogicalModel:Url
  </D2LogicalModel:photoUrl> [0..1] ?
  <D2LogicalModel:applicableForUser> D2LogicalModel:UserTypeEnum
  </D2LogicalModel:applicableForUser> [0..*] ?
  <D2LogicalModel:availabilityAndOpeningTimes>
    D2LogicalModel:OpeningTimes
  </D2LogicalModel:availabilityAndOpeningTimes> [0..1] ?
  <D2LogicalModel:tariffsAndPayment> D2LogicalModel:TariffsAndPayment
  </D2LogicalModel:tariffsAndPayment> [0..1]
  <D2LogicalModel:groupOfLocations> D2LogicalModel:GroupOfLocations
  </D2LogicalModel:groupOfLocations> [0..1]
  <D2LogicalModel:applicableForVehicles>
    D2LogicalModel:VehicleCharacteristics

```

```

</D2LogicalModel:applicableForVehicles> [0..*]
<D2LogicalModel:parkingEquipmentOrServiceFacilityExtension>
D2LogicalModel: _ExtensionType
</D2LogicalModel:parkingEquipmentOrServiceFacilityExtension> [0..1]
<D2LogicalModel:equipmentType> D2LogicalModel:EquipmentTypeEnum
</D2LogicalModel:equipmentType> [1] ?
<D2LogicalModel:electricCharging> D2LogicalModel:ElectricCharging
</D2LogicalModel:electricCharging> [0..1]
<D2LogicalModel:equipmentExtension> D2LogicalModel: _ExtensionType
</D2LogicalModel:equipmentExtension> [0..1]
</...>

```

Schema Component Representation

```

<xs:complexType name="Equipment">
  <xs:complexContent>
    <xs:extension
      base="D2LogicalModel:ParkingEquipmentOrServiceFacility">
      <xs:sequence>
        <xs:element name="equipmentType"
          type="D2LogicalModel:EquipmentTypeEnum" minOccurs="1"
          maxOccurs="1"/>
        <xs:element name="electricCharging"
          type="D2LogicalModel:ElectricCharging" minOccurs="0"/>
        <xs:element name="equipmentExtension"
          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
<u>Abstract</u>	no
Documentation	Details associated with the management of the exchange between the supplier and the client.

XML Instance Representation

```

<...>
  <D2LogicalModel:keepAlive> D2LogicalModel:Boolean
</D2LogicalModel:keepAlive> [0..1] ?
  <D2LogicalModel:response> D2LogicalModel:ResponseEnum
</D2LogicalModel:response> [0..1] ?
  <D2LogicalModel:subscriptionReference> D2LogicalModel:String
</D2LogicalModel:subscriptionReference> [0..1] ?
  <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="keepAlive" type="D2LogicalModel:Boolean"
      minOccurs="0" maxOccurs="1"/>
    <xs:element name="response" type="D2LogicalModel:ResponseEnum"
      minOccurs="0" maxOccurs="1"/>
    <xs:element name="subscriptionReference"
      type="D2LogicalModel:String" minOccurs="0" maxOccurs="1"/>
    <xs:element name="supplierIdentification"
      type="D2LogicalModel:InternationalIdentifier"/>
    <xs:element name="exchangeExtension"
      type="D2LogicalModel:ExtensionType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
```

[top](#)

Complex Type: ExternalReferencing

Super-types: None

Sub-types: None

Name	ExternalReferencing
Abstract	no
Documentation	A location defined by reference to an external/other referencing system.

XML Instance Representation

```
<...>
  <D2LogicalModel:externalLocationCode> D2LogicalModel:String
</D2LogicalModel:externalLocationCode> [1] ?
  <D2LogicalModel:externalReferencingSystem> D2LogicalModel:String
</D2LogicalModel:externalReferencingSystem> [1] ?
  <D2LogicalModel:externalReferencingExtension>
    D2LogicalModel:ExtensionType
  </D2LogicalModel:externalReferencingExtension> [0..1]
</...>
```

Schema Component Representation

```
<xs:complexType name="ExternalReferencing">
  <xs:sequence>
    <xs:element name="externalLocationCode"
      type="D2LogicalModel:String" minOccurs="1" maxOccurs="1"/>
    <xs:element name="externalReferencingSystem"
      type="D2LogicalModel:String" minOccurs="1" maxOccurs="1"/>
    <xs:element name="externalReferencingExtension"
      type="D2LogicalModel:ExtensionType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
```

[top](#)

Complex Type: GenericPublication

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

Sub-types: None

Name GenericPublication

Abstract no

Documentation A publication used to make level B extensions at the publication level.

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:genericPublicationName> D2LogicalModel:String  
    </D2LogicalModel:genericPublicationName> [1] ?  
    <D2LogicalModel:genericPublicationExtension>  
    D2LogicalModel:_GenericPublicationExtensionType  
    </D2LogicalModel:genericPublicationExtension> [0..1]  
</...>
```

Schema Component Representation

```
<xs:complexType name="GenericPublication">  
  <xs:complexContent>  
    <xs:extension base="D2LogicalModel:PayloadPublication">  
      <xs:sequence>  
        <xs:element name="genericPublicationName"  
          type="D2LogicalModel:String" minOccurs="1" maxOccurs="1"/>  
        <xs:element name="genericPublicationExtension"  
          type="D2LogicalModel:_GenericPublicationExtensionType"  
          minOccurs="0"/>  
      </xs:sequence>  
    </xs:extension>  
  </xs:complexContent>  
</xs:complexType>
```

[top](#)

Complex Type: **GrossWeightCharacteristic**

Super-types: None

Sub-types: None

Name GrossWeightCharacteristic

Abstract no

Documentation Gross weight characteristic of a vehicle.

XML Instance Representation

```
<...>
```

```

<D2LogicalModel:comparisonOperator>
D2LogicalModel:ComparisonOperatorEnum
</D2LogicalModel:comparisonOperator> [1] ?
<D2LogicalModel:grossVehicleWeight> D2LogicalModel:Tonnes
</D2LogicalModel:grossVehicleWeight> [1] ?
<D2LogicalModel:grossWeightCharacteristicExtension>
D2LogicalModel:ExtensionType
</D2LogicalModel:grossWeightCharacteristicExtension> [0..1]
</...>

```

Schema Component Representation

```

<xs:complexType name="GrossWeightCharacteristic">
  <xs:sequence>
    <xs:element name="comparisonOperator"
      type="D2LogicalModel:ComparisonOperatorEnum" minOccurs="1"
      maxOccurs="1"/>
    <xs:element name="grossVehicleWeight"
      type="D2LogicalModel:Tonnes" minOccurs="1" maxOccurs="1"/>
    <xs:element name="grossWeightCharacteristicExtension"
      type="D2LogicalModel:ExtensionType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>

```

[top](#)

Complex Type: **GroupOfLocations**

Super-types: None

Sub-types:

- [Itinerary](#) (by extension)
 - [ItineraryByIndexedLocations](#) (by extension)
 - [ItineraryByReference](#) (by extension)
- [Location](#) (by extension)
 - [Area](#) (by extension)
 - [LocationByReference](#) (by extension)
 - [NetworkLocation](#) (by extension)
 - [Linear](#) (by extension)
 - [Point](#) (by extension)
- [NonOrderedLocations](#) (by extension)
 - [NonOrderedLocationGroupByList](#) (by extension)
 - [NonOrderedLocationGroupByReference](#) (by extension)

Name	GroupOfLocations
Abstract	yes
Documentation	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: **GroupOfParkingSites**

Super-types: [ParkingRecord](#) < **GroupOfParkingSites** (by extension)

Sub-types: None

Name	GroupOfParkingSites
Abstract	no
Documentation	A logical composition of parking sites with aggregated properties (e.g. number of spaces). Examples: Urban parking area "West" or all truck parkings along a motorway. The included parking sites may -but must not- be specified as subcomponents.

XML Instance Representation

```
<...
  id="xs:string [1]"
  version="xs:string [1]">
  <D2LogicalModel:parkingName> D2LogicalModel:MultilingualString
</D2LogicalModel:parkingName> [1] ?
  <D2LogicalModel:parkingAlias> D2LogicalModel:MultilingualString
</D2LogicalModel:parkingAlias> [0..*] ?
  <D2LogicalModel:parkingDescription>
D2LogicalModel:MultilingualString
</D2LogicalModel:parkingDescription> [0..1] ?
  <D2LogicalModel:parkingRecordVersionTime> D2LogicalModel:DateTime
</D2LogicalModel:parkingRecordVersionTime> [1] ?
  <D2LogicalModel:parkingNumberOfSpaces>
D2LogicalModel:NonNegativeInteger
</D2LogicalModel:parkingNumberOfSpaces> [0..1] ?
  <D2LogicalModel:parkingPrincipalNumberOfSpaces>
D2LogicalModel:NonNegativeInteger
</D2LogicalModel:parkingPrincipalNumberOfSpaces> [0..1] ?
  <D2LogicalModel:maximumParkingDuration> D2LogicalModel:Seconds
</D2LogicalModel:maximumParkingDuration> [0..1] ?
  <D2LogicalModel:photoUrl> D2LogicalModel:Url
</D2LogicalModel:photoUrl> [0..1] ?
  <D2LogicalModel:urlLinkAddress> D2LogicalModel:Url
</D2LogicalModel:urlLinkAddress> [0..1] ?
  <D2LogicalModel:parkingOccupanyDetectionType>
D2LogicalModel:OccupancyDetectionTypeEnum
</D2LogicalModel:parkingOccupanyDetectionType> [0..*] ?
  <D2LogicalModel:emergencyContact> D2LogicalModel:Contact
</D2LogicalModel:emergencyContact> [0..*] ?
  <D2LogicalModel:owner> D2LogicalModel:Contact
</D2LogicalModel:owner> [0..*] ?
  <D2LogicalModel:responisbleAuthority> D2LogicalModel:Contact
</D2LogicalModel:responisbleAuthority> [0..*] ?
```

```

<D2LogicalModel:securityService> D2LogicalModel:Contact
</D2LogicalModel:securityService> [0..*] ?
<D2LogicalModel:operator> D2LogicalModel:Contact
</D2LogicalModel:operator> [1..*] ?
<D2LogicalModel:servicePartner> D2LogicalModel:Contact
</D2LogicalModel:servicePartner> [0..*] ?
<D2LogicalModel:parkingVMS> D2LogicalModel:ParkingVMS
</D2LogicalModel:parkingVMS> [0..*]
<D2LogicalModel:parkingLocation> D2LogicalModel:GroupOfLocations
</D2LogicalModel:parkingLocation> [1] ?
<D2LogicalModel:parkingRoute> D2LogicalModel:ParkingRoute
</D2LogicalModel:parkingRoute> [0..*]
<D2LogicalModel:parkingColour> D2LogicalModel:RGBColour
</D2LogicalModel:parkingColour> [0..1] ?
<D2LogicalModel:onlyAssignedParking>
D2LogicalModel:ParkingAssignment
</D2LogicalModel:onlyAssignedParking> [0..1] ?
<D2LogicalModel:assignedParkingAmongOthers>
D2LogicalModel:ParkingAssignment
</D2LogicalModel:assignedParkingAmongOthers> [0..1] ?
<D2LogicalModel:prohibitedParking> D2LogicalModel:ParkingAssignment
</D2LogicalModel:prohibitedParking> [0..1] ?
<D2LogicalModel:tariffsAndPayment> D2LogicalModel:TariffsAndPayment
</D2LogicalModel:tariffsAndPayment> [0..1]
<D2LogicalModel:parkingEquipmentOrServiceFacility>
D2LogicalModel: ParkingRecordEquipmentOrServiceFacilityIndexParkingEquipmen
</D2LogicalModel:parkingEquipmentOrServiceFacility> [0..*]
<D2LogicalModel:parkingSpace> D2LogicalModel: ParkingSpace
</D2LogicalModel:parkingSpace> [0..*] ?
<D2LogicalModel:groupOfParkingSpaces>
D2LogicalModel: GroupOfParkingSpaces
</D2LogicalModel:groupOfParkingSpaces> [0..*] ?
<D2LogicalModel:parkingThresholds> D2LogicalModel:ParkingThresholds
</D2LogicalModel:parkingThresholds> [0..1]
<D2LogicalModel:permitsAndProhibitions>
D2LogicalModel:PermitsAndProhibitions
</D2LogicalModel:permitsAndProhibitions> [0..*]
<D2LogicalModel:emergencyAssemblyPoint>
D2LogicalModel:GroupOfLocations
</D2LogicalModel:emergencyAssemblyPoint> [0..1] ?
<D2LogicalModel:entireArea> D2LogicalModel:Area
</D2LogicalModel:entireArea> [0..1] ?
<D2LogicalModel:parkingRecordDimension> D2LogicalModel:Dimension
</D2LogicalModel:parkingRecordDimension> [0..1] ?
<D2LogicalModel:parkingRecordExtension>
D2LogicalModel: ExtensionType
</D2LogicalModel:parkingRecordExtension> [0..1]
<D2LogicalModel:groupOfParkingSitesType>
D2LogicalModel:GroupOfParkingSitesTypeEnum
</D2LogicalModel:groupOfParkingSitesType> [0..1] ?
<D2LogicalModel:parkingSiteByReference>
D2LogicalModel: ParkingRecordVersionedReference
</D2LogicalModel:parkingSiteByReference> [0..*] ?
<D2LogicalModel:parkingSite> D2LogicalModel:ParkingSite
</D2LogicalModel:parkingSite> [0..*]
<D2LogicalModel:groupOfParkingSitesExtension>
D2LogicalModel: ExtensionType
</D2LogicalModel:groupOfParkingSitesExtension> [0..1]
</...>

```

Schema Component Representation

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

```

<xs:complexContent>
  <xs:extension base="D2LogicalModel:ParkingRecord">
    <xs:sequence>
      <xs:element name="groupOfParkingSitesType"
        type="D2LogicalModel:GroupOfParkingSitesTypeEnum"
        minOccurs="0" maxOccurs="1"/>
      <xs:element name="parkingSiteByReference"
        type="D2LogicalModel: ParkingRecordVersionedReference"
        minOccurs="0" maxOccurs="unbounded"/>
      <xs:element name="parkingSite"
        type="D2LogicalModel:ParkingSite" minOccurs="0"
        maxOccurs="unbounded"/>
      <xs:element name="groupOfParkingSitesExtension"
        type="D2LogicalModel: ExtensionType" minOccurs="0"/>
    </xs:sequence>
  </xs:extension>
</xs:complexContent>
</xs:complexType>

```

[top](#)

Complex Type: **GroupOfParkingSpaces**

Super-types:	ParkingSpaceBasics < GroupOfParkingSpaces (by extension)
Sub-types:	None

Name	GroupOfParkingSpaces
Abstract	no
Documentation	A group of parking spaces. All information provided has to be identical for all places in this group. Can also be used just to give the number of lorry parkings, for example. 'GroupOfParkingSpaces' may be multiple defined or include each other.

XML Instance Representation

```

<...>
  <D2LogicalModel:parkingSpaceOrGroupIdentifier>
    D2LogicalModel:MultilingualString
  </D2LogicalModel:parkingSpaceOrGroupIdentifier> [0..1] ?
  <D2LogicalModel:parkingFloorOrLevel> D2LogicalModel:Integer
  </D2LogicalModel:parkingFloorOrLevel> [0..1] ?
  <D2LogicalModel:accessibility> D2LogicalModel:AccessibilityEnum
  </D2LogicalModel:accessibility> [0..*] ?
  <D2LogicalModel:parkingSpaceAccessibility>
    D2LogicalModel:ParkingSpaceAccessibilityEnum
  </D2LogicalModel:parkingSpaceAccessibility> [0..*] ?
  <D2LogicalModel:parkingSpacePhysics>
    D2LogicalModel:ParkingSpacePhysicsEnum
  </D2LogicalModel:parkingSpacePhysics> [0..2] ?
  <D2LogicalModel:parkingMode> D2LogicalModel:ParkingModeEnum
  </D2LogicalModel:parkingMode> [0..1] ?
  <D2LogicalModel:parkingReservation>
    D2LogicalModel:ReservationTypeEnum
  </D2LogicalModel:parkingReservation> [0..1] ?
  <D2LogicalModel:maximumParkingDuration> D2LogicalModel:Seconds
  </D2LogicalModel:maximumParkingDuration> [0..1] ?
  <D2LogicalModel:distanceFromPrimaryRoad>
    D2LogicalModel:MetresAsNonNegativeInteger

```

```

</D2LogicalModel:distanceFromPrimaryRoad> [0..1] ?
<D2LogicalModel:parkingOccupancyDetectionType>
D2LogicalModel:OccupancyDetectionTypeEnum
</D2LogicalModel:parkingOccupancyDetectionType> [0..*] ?
<D2LogicalModel:parkingSecurity> D2LogicalModel:ParkingSecurityEnum
</D2LogicalModel:parkingSecurity> [0..*] ?
<D2LogicalModel:dedicatedAccess> D2LogicalModel:DedicatedAccess
</D2LogicalModel:dedicatedAccess> [0..*]
<D2LogicalModel:onlyAssignedParking>
D2LogicalModel:ParkingAssignment
</D2LogicalModel:onlyAssignedParking> [0..1] ?
<D2LogicalModel:assignedParkingAmongOthers>
D2LogicalModel:ParkingAssignment
</D2LogicalModel:assignedParkingAmongOthers> [0..1] ?
<D2LogicalModel:prohibitedParking> D2LogicalModel:ParkingAssignment
</D2LogicalModel:prohibitedParking> [0..1] ?
<D2LogicalModel:parkingEquipmentOrServiceFacility>
D2LogicalModel: ParkingSpaceBasicsEquipmentOrServiceFacilityIndexParkingEqu:
</D2LogicalModel:parkingEquipmentOrServiceFacility> [0..*] ?
<D2LogicalModel:parkingUsageScenario>
D2LogicalModel: ParkingSpaceBasicsScenarioIndexParkingUsageScenario
</D2LogicalModel:parkingUsageScenario> [0..*]
<D2LogicalModel:parkingSpaceBasicsExtension>
D2LogicalModel: ExtensionType
</D2LogicalModel:parkingSpaceBasicsExtension> [0..1]
<D2LogicalModel:parkingNumberOfSpaces>
D2LogicalModel:NonNegativeInteger
</D2LogicalModel:parkingNumberOfSpaces> [1] ?
<D2LogicalModel:parkingTypeOfGroup>
D2LogicalModel:ParkingTypeOfGroup
</D2LogicalModel:parkingTypeOfGroup> [1] ?
<D2LogicalModel:identicalToGroup> D2LogicalModel:IndexReference
</D2LogicalModel:identicalToGroup> [0..*] ?
<D2LogicalModel:realSubsetOfGroup> D2LogicalModel:IndexReference
</D2LogicalModel:realSubsetOfGroup> [0..*] ?
<D2LogicalModel:minimumParkingSpaceDimension>
D2LogicalModel:Dimension
</D2LogicalModel:minimumParkingSpaceDimension> [0..1] ?
<D2LogicalModel:dimensionOfGroup> D2LogicalModel:Dimension
</D2LogicalModel:dimensionOfGroup> [0..1] ?
<D2LogicalModel:maximumParkingSpaceDimension>
D2LogicalModel:Dimension
</D2LogicalModel:maximumParkingSpaceDimension> [0..1] ?
<D2LogicalModel:parkingSpace>
D2LogicalModel: GroupOfParkingSpacesParkingSpaceIndexParkingSpace
</D2LogicalModel:parkingSpace> [0..*]
<D2LogicalModel:groupOfLocations> D2LogicalModel:GroupOfLocations
</D2LogicalModel:groupOfLocations> [0..1]
<D2LogicalModel:groupOfParkingSpacesExtension>
D2LogicalModel: ExtensionType
</D2LogicalModel:groupOfParkingSpacesExtension> [0..1]
</...>

```

Schema Component Representation

```

<xs:complexType name="GroupOfParkingSpaces">
  <xs:complexContent>
    <xs:extension base="D2LogicalModel:ParkingSpaceBasics">
      <xs:sequence>
        <xs:element name="parkingNumberOfSpaces"
          type="D2LogicalModel:NonNegativeInteger" minOccurs="1"
          maxOccurs="1"/>
        <xs:element name="parkingTypeOfGroup"

```

```

type="D2LogicalModel: ParkingTypeOfGroup" minOccurs="1"
maxOccurs="1"/>
<xs:element name="identicalToGroup"
type="D2LogicalModel: IndexReference" minOccurs="0"
maxOccurs="unbounded"/>
<xs:element name="realSubsetOfGroup"
type="D2LogicalModel: IndexReference" minOccurs="0"
maxOccurs="unbounded"/>
<xs:element name="minimumParkingSpaceDimension"
type="D2LogicalModel: Dimension" minOccurs="0"/>
<xs:element name="dimensionOfGroup"
type="D2LogicalModel: Dimension" minOccurs="0"/>
<xs:element name="maximumParkingSpaceDimension"
type="D2LogicalModel: Dimension" minOccurs="0"/>
<xs:element name="parkingSpace"
type="D2LogicalModel: _GroupOfParkingSpacesParkingSpaceIndexParkingSp
minOccurs="0" maxOccurs="unbounded"/>
<xs:element name="groupOfLocations"
type="D2LogicalModel: GroupOfLocations" minOccurs="0"/>
<xs:element name="groupOfParkingSpacesExtension"
type="D2LogicalModel: _ExtensionType" minOccurs="0"/>
</xs:sequence>
</xs:extension>
</xs:complexContent>
</xs:complexType>

```

[top](#)

Complex Type: HazardousMaterials

Super-types: None

Sub-types: None

Name	HazardousMaterials
Abstract	no
Documentation	Details of hazardous materials.

XML Instance Representation

```

<...>
  <D2LogicalModel:chemicalName> D2LogicalModel:MultilingualString
</D2LogicalModel:chemicalName> [1] ?
  <D2LogicalModel:dangerousGoodsFlashPoint>
D2LogicalModel:TemperatureCelsius
</D2LogicalModel:dangerousGoodsFlashPoint> [0..1] ?
  <D2LogicalModel:dangerousGoodsRegulations>
D2LogicalModel:DangerousGoodsRegulationsEnum
</D2LogicalModel:dangerousGoodsRegulations> [0..1] ?
  <D2LogicalModel:hazardCodeIdentification> D2LogicalModel:String
</D2LogicalModel:hazardCodeIdentification> [0..1] ?
  <D2LogicalModel:hazardCodeVersionNumber>
D2LogicalModel:NonNegativeInteger
</D2LogicalModel:hazardCodeVersionNumber> [0..1] ?
  <D2LogicalModel:hazardSubstanceItemPageNumber>
D2LogicalModel:String
</D2LogicalModel:hazardSubstanceItemPageNumber> [0..1] ?
  <D2LogicalModel:tremCardNumber> D2LogicalModel:String
</D2LogicalModel:tremCardNumber> [0..1] ?
  <D2LogicalModel:undgNumber> D2LogicalModel:String

```

```

</D2LogicalModel:undgNumber> [0..1] ?
<D2LogicalModel:volumeOfDangerousGoods> D2LogicalModel:CubicMetres
</D2LogicalModel:volumeOfDangerousGoods> [0..1] ?
<D2LogicalModel:weightOfDangerousGoods> D2LogicalModel:Tonnes
</D2LogicalModel:weightOfDangerousGoods> [0..1] ?
<D2LogicalModel:hazardousMaterialsExtension>
D2LogicalModel: ExtensionType
</D2LogicalModel:hazardousMaterialsExtension> [0..1]
</...>

```

Schema Component Representation

```

<xs:complexType name="HazardousMaterials">
  <xs:sequence>
    <xs:element name="chemicalName"
      type="D2LogicalModel:MultilingualString" minOccurs="1"
      maxOccurs="1"/>
    <xs:element name="dangerousGoodsFlashPoint"
      type="D2LogicalModel:TemperatureCelsius" minOccurs="0"
      maxOccurs="1"/>
    <xs:element name="dangerousGoodsRegulations"
      type="D2LogicalModel:DangerousGoodsRegulationsEnum"
      minOccurs="0" maxOccurs="1"/>
    <xs:element name="hazardCodeIdentification"
      type="D2LogicalModel:String" minOccurs="0" maxOccurs="1"/>
    <xs:element name="hazardCodeVersionNumber"
      type="D2LogicalModel:NonNegativeInteger" minOccurs="0"
      maxOccurs="1"/>
    <xs:element name="hazardSubstanceItemPageNumber"
      type="D2LogicalModel:String" minOccurs="0" maxOccurs="1"/>
    <xs:element name="tremCardNumber" type="D2LogicalModel:String"
      minOccurs="0" maxOccurs="1"/>
    <xs:element name="undgNumber" type="D2LogicalModel:String"
      minOccurs="0" maxOccurs="1"/>
    <xs:element name="volumeOfDangerousGoods"
      type="D2LogicalModel:CubicMetres" minOccurs="0" maxOccurs="1"/>
    <xs:element name="weightOfDangerousGoods"
      type="D2LogicalModel:Tonnes" minOccurs="0" maxOccurs="1"/>
    <xs:element name="hazardousMaterialsExtension"
      type="D2LogicalModel: ExtensionType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>

```

[top](#)

Complex Type: **HeaderInformation**

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

Name	HeaderInformation
<u>Abstract</u>	no
Documentation	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: **HeaviestAxleWeightCharacteristic**

Super-types:	None
Sub-types:	None

Name	HeaviestAxleWeightCharacteristic
Abstract	no
Documentation	Weight characteristic of the heaviest axle on the vehicle.

XML Instance Representation

```

<...>
  <D2LogicalModel:comparisonOperator>
  D2LogicalModel:ComparisonOperatorEnum
  </D2LogicalModel:comparisonOperator> [1] ?
  <D2LogicalModel:heaviestAxleWeight> D2LogicalModel:Tonnes
  </D2LogicalModel:heaviestAxleWeight> [1] ?
  <D2LogicalModel:heaviestAxleWeightCharacteristicExtension>
  D2LogicalModel: _ExtensionType
  </D2LogicalModel:heaviestAxleWeightCharacteristicExtension> [0..1]
</...>

```

Schema Component Representation

```

<xs:complexType name="HeaviestAxleWeightCharacteristic">
  <xs:sequence>
    <xs:element name="comparisonOperator"
      type="D2LogicalModel:ComparisonOperatorEnum" minOccurs="1"
      maxOccurs="1"/>
    <xs:element name="heaviestAxleWeight"
      type="D2LogicalModel:Tonnes" minOccurs="1" maxOccurs="1"/>
    <xs:element name="heaviestAxleWeightCharacteristicExtension"
      type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>

```

[top](#)

Complex Type: **HeightCharacteristic**

Super-types: None

Sub-types: None

Name	HeightCharacteristic
<u>Abstract</u>	no
Documentation	Height characteristic of a vehicle.

XML Instance Representation

```

<...>
  <D2LogicalModel:comparisonOperator>
    D2LogicalModel:ComparisonOperatorEnum
  </D2LogicalModel:comparisonOperator> [1] ?
  <D2LogicalModel:vehicleHeight> D2LogicalModel:MetresAsFloat
  </D2LogicalModel:vehicleHeight> [1] ?
  <D2LogicalModel:heightCharacteristicExtension>
    D2LogicalModel:_ExtensionType
  </D2LogicalModel:heightCharacteristicExtension> [0..1]
</...>

```

Schema Component Representation

```

<xs:complexType name="HeightCharacteristic">
  <xs:sequence>
    <xs:element name="comparisonOperator"
      type="D2LogicalModel:ComparisonOperatorEnum" minOccurs="1"
      maxOccurs="1"/>
    <xs:element name="vehicleHeight"
      type="D2LogicalModel:MetresAsFloat" minOccurs="1"
      maxOccurs="1"/>
    <xs:element name="heightCharacteristicExtension"
      type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>

```

[top](#)

Complex Type: **InterUrbanParkingSite**

Super-types:	ParkingRecord < ParkingSite (by extension) < InterUrbanParkingSite (by extension)
--------------	--

Sub-types: None

Name InterUrbanParkingSite
Abstract no
Documentation A parking site in an interurban context.

XML Instance Representation

```
<...  
  id="xs:string [1]"  
  version="xs:string [1]">  
    <D2LogicalModel:parkingName> D2LogicalModel:MultilingualString  
    </D2LogicalModel:parkingName> [1] ?  
    <D2LogicalModel:parkingAlias> D2LogicalModel:MultilingualString  
    </D2LogicalModel:parkingAlias> [0..*] ?  
    <D2LogicalModel:parkingDescription>  
    D2LogicalModel:MultilingualString  
    </D2LogicalModel:parkingDescription> [0..1] ?  
    <D2LogicalModel:parkingRecordVersionTime> D2LogicalModel:DateTime  
    </D2LogicalModel:parkingRecordVersionTime> [1] ?  
    <D2LogicalModel:parkingNumberOfSpaces>  
    D2LogicalModel:NonNegativeInteger  
    </D2LogicalModel:parkingNumberOfSpaces> [0..1] ?  
    <D2LogicalModel:parkingPrincipalNumberOfSpaces>  
    D2LogicalModel:NonNegativeInteger  
    </D2LogicalModel:parkingPrincipalNumberOfSpaces> [0..1] ?  
    <D2LogicalModel:maximumParkingDuration> D2LogicalModel:Seconds  
    </D2LogicalModel:maximumParkingDuration> [0..1] ?  
    <D2LogicalModel:photoUrl> D2LogicalModel:Url  
    </D2LogicalModel:photoUrl> [0..1] ?  
    <D2LogicalModel:urlLinkAddress> D2LogicalModel:Url  
    </D2LogicalModel:urlLinkAddress> [0..1] ?  
    <D2LogicalModel:parkingOccupanyDetectionType>  
    D2LogicalModel:OccupancyDetectionTypeEnum  
    </D2LogicalModel:parkingOccupanyDetectionType> [0..*] ?  
    <D2LogicalModel:emergencyContact> D2LogicalModel:Contact  
    </D2LogicalModel:emergencyContact> [0..*] ?  
    <D2LogicalModel:owner> D2LogicalModel:Contact  
    </D2LogicalModel:owner> [0..*] ?  
    <D2LogicalModel:responisbleAuthority> D2LogicalModel:Contact  
    </D2LogicalModel:responisbleAuthority> [0..*] ?  
    <D2LogicalModel:securityService> D2LogicalModel:Contact  
    </D2LogicalModel:securityService> [0..*] ?  
    <D2LogicalModel:operator> D2LogicalModel:Contact  
    </D2LogicalModel:operator> [1..*] ?  
    <D2LogicalModel:servicePartner> D2LogicalModel:Contact  
    </D2LogicalModel:servicePartner> [0..*] ?  
    <D2LogicalModel:parkingVMS> D2LogicalModel:ParkingVMS  
    </D2LogicalModel:parkingVMS> [0..*]  
    <D2LogicalModel:parkingLocation> D2LogicalModel:GroupOfLocations  
    </D2LogicalModel:parkingLocation> [1] ?  
    <D2LogicalModel:parkingRoute> D2LogicalModel:ParkingRoute  
    </D2LogicalModel:parkingRoute> [0..*]  
    <D2LogicalModel:parkingColour> D2LogicalModel:RGBColour  
    </D2LogicalModel:parkingColour> [0..1] ?  
    <D2LogicalModel:onlyAssignedParking>  
    D2LogicalModel:ParkingAssignment  
    </D2LogicalModel:onlyAssignedParking> [0..1] ?  
    <D2LogicalModel:assignedParkingAmongOthers>  
    D2LogicalModel:ParkingAssignment
```

```

</D2LogicalModel:assignedParkingAmongOthers> [0..1] ?
<D2LogicalModel:prohibitedParking> D2LogicalModel:ParkingAssignment
</D2LogicalModel:prohibitedParking> [0..1] ?
<D2LogicalModel:tariffsAndPayment> D2LogicalModel:TariffsAndPayment
</D2LogicalModel:tariffsAndPayment> [0..1]
<D2LogicalModel:parkingEquipmentOrServiceFacility>
D2LogicalModel: ParkingRecordEquipmentOrServiceFacilityIndexParkingEquipmen
</D2LogicalModel:parkingEquipmentOrServiceFacility> [0..*]
<D2LogicalModel:parkingSpace> D2LogicalModel: ParkingSpace
</D2LogicalModel:parkingSpace> [0..*] ?
<D2LogicalModel:groupOfParkingSpaces>
D2LogicalModel: GroupOfParkingSpaces
</D2LogicalModel:groupOfParkingSpaces> [0..*] ?
<D2LogicalModel:parkingThresholds> D2LogicalModel:ParkingThresholds
</D2LogicalModel:parkingThresholds> [0..1]
<D2LogicalModel:permitsAndProhibitions>
D2LogicalModel:PermitsAndProhibitions
</D2LogicalModel:permitsAndProhibitions> [0..*]
<D2LogicalModel:emergencyAssemblyPoint>
D2LogicalModel:GroupOfLocations
</D2LogicalModel:emergencyAssemblyPoint> [0..1] ?
<D2LogicalModel:entireArea> D2LogicalModel:Area
</D2LogicalModel:entireArea> [0..1] ?
<D2LogicalModel:parkingRecordDimension> D2LogicalModel:Dimension
</D2LogicalModel:parkingRecordDimension> [0..1] ?
<D2LogicalModel:parkingRecordExtension>
D2LogicalModel: ExtensionType
</D2LogicalModel:parkingRecordExtension> [0..1]
<D2LogicalModel:parkingReservation>
D2LogicalModel:ReservationTypeEnum
</D2LogicalModel:parkingReservation> [0..1] ?
<D2LogicalModel:parkingLayout> D2LogicalModel:ParkingLayoutEnum
</D2LogicalModel:parkingLayout> [0..*] ?
<D2LogicalModel:highestFloor> D2LogicalModel:Integer
</D2LogicalModel:highestFloor> [0..1] ?
<D2LogicalModel:lowestFloor> D2LogicalModel:Integer
</D2LogicalModel:lowestFloor> [0..1] ?
<D2LogicalModel:temporaryParking> D2LogicalModel:Boolean
</D2LogicalModel:temporaryParking> [0..1] ?
<D2LogicalModel:parkingSiteAddress> D2LogicalModel:Contact
</D2LogicalModel:parkingSiteAddress> [1..*] ?
<D2LogicalModel:reservationService> D2LogicalModel:Contact
</D2LogicalModel:reservationService> [0..*] ?
<D2LogicalModel:parkingUsageScenario>
D2LogicalModel: ParkingSiteScenarioIndexParkingUsageScenario
</D2LogicalModel:parkingUsageScenario> [0..*]
<D2LogicalModel:openingTimes> D2LogicalModel:OpeningTimes
</D2LogicalModel:openingTimes> [0..1]
<D2LogicalModel:parkingAccess> D2LogicalModel:ParkingAccess
</D2LogicalModel:parkingAccess> [1..*] ?
<D2LogicalModel:parkingStandardsAndSecurity>
D2LogicalModel:ParkingStandardsAndSecurity
</D2LogicalModel:parkingStandardsAndSecurity> [1]
<D2LogicalModel:parkingSiteExtension> D2LogicalModel: ExtensionType
</D2LogicalModel:parkingSiteExtension> [0..1]
<D2LogicalModel:interUrbanParkingSiteLocation>
D2LogicalModel: InterUrbanParkingSiteLocationEnum
</D2LogicalModel:interUrbanParkingSiteLocation> [1] ?
<D2LogicalModel:interUrbanParkingSiteExtension>
D2LogicalModel: ExtensionType
</D2LogicalModel:interUrbanParkingSiteExtension> [0..1]
</...>

```

Schema Component Representation

```
<xs:complexType name="InterUrbanParkingSite">
  <xs:complexContent>
    <xs:extension base="D2LogicalModel:ParkingSite">
      <xs:sequence>
        <xs:element name="interUrbanParkingSiteLocation"
          type="D2LogicalModel:InterUrbanParkingSiteLocationEnum"
          minOccurs="1" maxOccurs="1"/>
        <xs:element name="interUrbanParkingSiteExtension"
          type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

[top](#)

Complex Type: **InternationalIdentifier**

Super-types:	None
Sub-types:	None

Name	InternationalIdentifier
Abstract	no
Documentation	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:internationalIdentifier>
    D2LogicalModel:_ExtensionType
  </D2LogicalModel:internationalIdentifier> [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: **Itinerary**

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

Sub-types:

- [ItineraryByIndexedLocations](#) (by extension)
- [ItineraryByReference](#) (by extension)

Name	Itinerary
Abstract	yes
Documentation	Multiple (i.e. more than one) physically separate locations arranged as an ordered set that defines an itinerary or route.

XML Instance Representation

```
<...>
  <D2LogicalModel:groupOfLocationsExtension>
    D2LogicalModel: _ExtensionType
  </D2LogicalModel:groupOfLocationsExtension> [0..1]
  <D2LogicalModel:routeDestination> D2LogicalModel:Destination
</D2LogicalModel:routeDestination> [0..*] ?
  <D2LogicalModel:itineraryExtension> D2LogicalModel: _ExtensionType
</D2LogicalModel:itineraryExtension> [0..1]
</...>
```

Schema Component Representation

```
<xs:complexType name="Itinerary" abstract="true">
  <xs:complexContent>
    <xs:extension base="D2LogicalModel:GroupOfLocations">
      <xs:sequence>
        <xs:element name="routeDestination"
          type="D2LogicalModel:Destination" minOccurs="0"
          maxOccurs="unbounded"/>
        <xs:element name="itineraryExtension"
          type="D2LogicalModel: _ExtensionType" minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

[top](#)

Complex Type: **ItineraryByIndexedLocations**

Super-types: [GroupOfLocations](#) < [Itinerary](#) (by extension) < **ItineraryByIndexedLocations** (by extension)

Sub-types: None

Name	ItineraryByIndexedLocations
Abstract	no
Documentation	Multiple physically separate locations arranged as an ordered set that defines an itinerary or route. The index qualifier indicates the order.

XML Instance Representation

```
<...>
  <D2LogicalModel:groupOfLocationsExtension>
    D2LogicalModel: _ExtensionType
```

```

</D2LogicalModel:groupOfLocationsExtension> [0..1]
<D2LogicalModel:routeDestination> D2LogicalModel:Destination
</D2LogicalModel:routeDestination> [0..*] ?
<D2LogicalModel:itineraryExtension> D2LogicalModel: _ExtensionType
</D2LogicalModel:itineraryExtension> [0..1]
<D2LogicalModel:locationContainedInItinerary>
D2LogicalModel: _LocationContainedInItinerary
</D2LogicalModel:locationContainedInItinerary> [0..*] ?
<D2LogicalModel:itineraryByIndexedLocationsExtension>
D2LogicalModel: _ExtensionType
</D2LogicalModel:itineraryByIndexedLocationsExtension> [0..1]
</...>

```

Schema Component Representation

```

<xs:complexType name="ItineraryByIndexedLocations">
  <xs:complexContent>
    <xs:extension base="D2LogicalModel:Itinerary">
      <xs:sequence>
        <xs:element name="locationContainedInItinerary"
          type="D2LogicalModel: _LocationContainedInItinerary"
          minOccurs="0" maxOccurs="unbounded"/>
        <xs:element name="itineraryByIndexedLocationsExtension"
          type="D2LogicalModel: _ExtensionType" minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

```

[top](#)

Complex Type: **ItineraryByReference**

Super-types: [GroupOfLocations](#) < [Itinerary](#) (by extension) < **ItineraryByReference** (by extension)

Sub-types: None

Name	ItineraryByReference
Abstract	no
Documentation	Multiple (i.e. more than one) physically separate locations which are ordered that constitute an itinerary or route where they are defined by reference to a predefined itinerary.

XML Instance Representation

```

<...>
  <D2LogicalModel:groupOfLocationsExtension>
    D2LogicalModel: _ExtensionType
  </D2LogicalModel:groupOfLocationsExtension> [0..1]
  <D2LogicalModel:routeDestination> D2LogicalModel:Destination
  </D2LogicalModel:routeDestination> [0..*] ?
  <D2LogicalModel:itineraryExtension> D2LogicalModel: _ExtensionType
  </D2LogicalModel:itineraryExtension> [0..1]
  <D2LogicalModel:predefinedItineraryReference>
    D2LogicalModel: _PredefinedItineraryVersionedReference
  </D2LogicalModel:predefinedItineraryReference> [1] ?
  <D2LogicalModel:itineraryByReferenceExtension>
    D2LogicalModel: _ExtensionType
  </D2LogicalModel:itineraryByReferenceExtension> [0..1]

```

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

Schema Component Representation

```
<xs:complexType name="ItineraryByReference">
  <xs:complexContent>
    <xs:extension base="D2LogicalModel:Itinerary">
      <xs:sequence>
        <xs:element name="predefinedItineraryReference"
          type="D2LogicalModel:PredefinedItineraryVersionedReference"
          minOccurs="1" maxOccurs="1"/>
        <xs:element name="itineraryByReferenceExtension"
          type="D2LogicalModel:ExtensionType" minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

[top](#)

Complex Type: **Junction**

Super-types: None

Sub-types: None

Name	Junction
<u>Abstract</u>	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:sequence>
</xs:complexType>
```

```

<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: **LengthCharacteristic**

Super-types:	None
Sub-types:	None

Name	LengthCharacteristic
Abstract	no
Documentation	Length characteristic of a vehicle.

XML Instance Representation

```

<...>
  <D2LogicalModel:comparisonOperator>
    D2LogicalModel:ComparisonOperatorEnum
  </D2LogicalModel:comparisonOperator> [1] ?
  <D2LogicalModel:vehicleLength> D2LogicalModel:MetresAsFloat
  </D2LogicalModel:vehicleLength> [1] ?
  <D2LogicalModel:lengthCharacteristicExtension>
    D2LogicalModel:ExtensionType
  </D2LogicalModel:lengthCharacteristicExtension> [0..1]
</...>

```

Schema Component Representation

```

<xs:complexType name="LengthCharacteristic">
  <xs:sequence>
    <xs:element name="comparisonOperator"
      type="D2LogicalModel:ComparisonOperatorEnum" minOccurs="1"
      maxOccurs="1"/>
    <xs:element name="vehicleLength"
      type="D2LogicalModel:MetresAsFloat" minOccurs="1"
      maxOccurs="1"/>
    <xs:element name="lengthCharacteristicExtension"
      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:externalReferencing>
    D2LogicalModel:ExternalReferencing
  </D2LogicalModel:externalReferencing> [0..*]
  <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:destination> D2LogicalModel:Destination
  </D2LogicalModel:destination> [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:_ExtensionType
  </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:_ExtensionType" minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

[top](#)

Complex Type: **LinearElement**

Super-types: None

Sub-types:

- [LinearElementByCode](#) (by extension)
- [LinearElementByPoints](#) (by extension)

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: [LinearElement](#) < **LinearElementByCode** (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: [LinearElement](#) < LinearElementByPoints (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>
```

Complex Type: **Location**

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

Sub-types:

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

Name	Location
<u>Abstract</u>	yes
Documentation	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:externalReferencing>
    D2LogicalModel:ExternalReferencing
  </D2LogicalModel:externalReferencing> [0..*]
  <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="externalReferencing"
          type="D2LogicalModel:ExternalReferencing" minOccurs="0"
          maxOccurs="unbounded"/>
        <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>
```

Complex Type: **LocationByReference**

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

Sub-types: None

Name LocationByReference
Abstract no
Documentation A location defined by reference to a predefined location.

XML Instance Representation

```
<...>
  <D2LogicalModel:groupOfLocationsExtension>
    D2LogicalModel:_ExtensionType
  </D2LogicalModel:groupOfLocationsExtension> [0..1]
  <D2LogicalModel:externalReferencing>
    D2LogicalModel:ExternalReferencing
  </D2LogicalModel:externalReferencing> [0..*]
  <D2LogicalModel:locationForDisplay> D2LogicalModel:PointCoordinates
</D2LogicalModel:locationForDisplay> [0..1] ?
  <D2LogicalModel:locationExtension> D2LogicalModel:_ExtensionType
</D2LogicalModel:locationExtension> [0..1]
  <D2LogicalModel:predefinedLocationReference>
    D2LogicalModel:_PredefinedLocationVersionedReference
  </D2LogicalModel:predefinedLocationReference> [1] ?
  <D2LogicalModel:locationByReferenceExtension>
    D2LogicalModel:_ExtensionType
  </D2LogicalModel:locationByReferenceExtension> [0..1]
</...>
```

Schema Component Representation

```
<xs:complexType name="LocationByReference">
  <xs:complexContent>
    <xs:extension base="D2LogicalModel:Location">
      <xs:sequence>
        <xs:element name="predefinedLocationReference"
          type="D2LogicalModel:_PredefinedLocationVersionedReference"
          minOccurs="1" maxOccurs="1"/>
        <xs:element name="locationByReferenceExtension"
          type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

[top](#)

Complex Type: MultilingualString

Super-types: None

Sub-types: None

Name MultilingualString
Abstract no

XML Instance Representation

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

```

    </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](#) < [MultilingualStringValue](#) (by restriction) < [MultilingualStringValue](#) (by extension)

Sub-types: None

Name MultilingualStringValue

Abstract no

XML Instance Representation

```

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

```

Schema Component Representation

```

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

```

[top](#)

Complex Type: NamedArea

Super-types: None

Sub-types: None

Name NamedArea

Abstract

no

Documentation

An area defined by a name and/or in terms of known boundaries, such as country or county boundaries or allocated control area of particular authority. The attributes do not form a union; instead, the smallest intersection forms the resulting area.

XML Instance Representation

```
<...>
  <D2LogicalModel:country> D2LogicalModel:CountryEnum
</D2LogicalModel:country> [0..1] ?
  <D2LogicalModel:nation> D2LogicalModel:MultilingualString
</D2LogicalModel:nation> [0..1] ?
  <D2LogicalModel:county> D2LogicalModel:MultilingualString
</D2LogicalModel:county> [0..1] ?
  <D2LogicalModel:areaName> D2LogicalModel:MultilingualString
</D2LogicalModel:areaName> [0..1] ?
  <D2LogicalModel:policeForceControlArea>
    D2LogicalModel:MultilingualString
  </D2LogicalModel:policeForceControlArea> [0..1] ?
  <D2LogicalModel:roadOperatorControlArea>
    D2LogicalModel:MultilingualString
  </D2LogicalModel:roadOperatorControlArea> [0..1] ?
  <D2LogicalModel:namedAreaExtension> D2LogicalModel:_ExtensionType
</D2LogicalModel:namedAreaExtension> [0..1]
</...>
```

Schema Component Representation

```
<xs:complexType name="NamedArea">
  <xs:sequence>
    <xs:element name="country" type="D2LogicalModel:CountryEnum"
      minOccurs="0" maxOccurs="1"/>
    <xs:element name="nation"
      type="D2LogicalModel:MultilingualString" minOccurs="0"
      maxOccurs="1"/>
    <xs:element name="county"
      type="D2LogicalModel:MultilingualString" minOccurs="0"
      maxOccurs="1"/>
    <xs:element name="areaName"
      type="D2LogicalModel:MultilingualString" minOccurs="0"
      maxOccurs="1"/>
    <xs:element name="policeForceControlArea"
      type="D2LogicalModel:MultilingualString" minOccurs="0"
      maxOccurs="1"/>
    <xs:element name="roadOperatorControlArea"
      type="D2LogicalModel:MultilingualString" minOccurs="0"
      maxOccurs="1"/>
    <xs:element name="namedAreaExtension"
      type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
  </xs:sequence>
</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:externalReferencing>
    D2LogicalModel:ExternalReferencing
  </D2LogicalModel:externalReferencing> [0..*]
  <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:destination> D2LogicalModel:Destination
  </D2LogicalModel:destination> [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="destination"
          type="D2LogicalModel:Destination" minOccurs="0"/>
        <xs:element name="networkLocationExtension"
          type="D2LogicalModel: _ExtensionType" minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

[top](#)

Complex Type: **NonOrderedLocationGroupByList**

Super-types: [GroupOfLocations](#) < [NonOrderedLocations](#) (by extension) < **NonOrderedLocationGroupByList** (by extension)

Sub-types: None

Name	NonOrderedLocationGroupByList
Abstract	no
Documentation	A group of (i.e. more than one) physically separate locations which have no specific order and where each location is explicitly listed.

XML Instance Representation

```

<...>
  <D2LogicalModel:groupOfLocationsExtension>
    D2LogicalModel: _ExtensionType
  </D2LogicalModel:groupOfLocationsExtension> [0..1]
  <D2LogicalModel:nonOrderedLocationsExtension>
    D2LogicalModel: _ExtensionType
  </D2LogicalModel:nonOrderedLocationsExtension> [0..1]
  <D2LogicalModel:locationContainedInGroup> D2LogicalModel:Location
  </D2LogicalModel:locationContainedInGroup> [2..*] ?
  <D2LogicalModel:nonOrderedLocationGroupByListExtension>
    D2LogicalModel: _ExtensionType
  </D2LogicalModel:nonOrderedLocationGroupByListExtension> [0..1]
</...>

```

Schema Component Representation

```

<xs:complexType name="NonOrderedLocationGroupByList">
  <xs:complexContent>
    <xs:extension base="D2LogicalModel:NonOrderedLocations">
      <xs:sequence>
        <xs:element name="locationContainedInGroup"
          type="D2LogicalModel:Location" minOccurs="2"
          maxOccurs="unbounded"/>
        <xs:element name="nonOrderedLocationGroupByListExtension"
          type="D2LogicalModel: _ExtensionType" minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

```

[top](#)

Complex Type: NonOrderedLocationGroupByReference

Super-types: [GroupOfLocations](#) < [NonOrderedLocations](#) (by extension) < **NonOrderedLocationGroupByReference** (by extension)

Sub-types: None

Name	NonOrderedLocationGroupByReference
Abstract	no
Documentation	A group of (i.e. more than one) physically separate locations which have no specific order that are defined by reference to a predefined non ordered location group.

XML Instance Representation

```

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

```

```

<D2LogicalModel:nonOrderedLocationsExtension>
D2LogicalModel: ExtensionType
</D2LogicalModel:nonOrderedLocationsExtension> [0..1]
<D2LogicalModel:predefinedNonOrderedLocationGroupReference>
D2LogicalModel: PredefinedNonOrderedLocationGroupVersionedReference
</D2LogicalModel:predefinedNonOrderedLocationGroupReference> [1] ?
<D2LogicalModel:nonOrderedLocationGroupByReferenceExtension>
D2LogicalModel: ExtensionType
</D2LogicalModel:nonOrderedLocationGroupByReferenceExtension>
[0..1]
</...>

```

Schema Component Representation

```

<xs:complexType name="NonOrderedLocationGroupByReference">
  <xs:complexContent>
    <xs:extension base="D2LogicalModel:NonOrderedLocations">
      <xs:sequence>
        <xs:element
          name="predefinedNonOrderedLocationGroupReference"
          type="D2LogicalModel: PredefinedNonOrderedLocationGroupVersionedReference"
          minOccurs="1" maxOccurs="1"/>
        <xs:element
          name="nonOrderedLocationGroupByReferenceExtension"
          type="D2LogicalModel: ExtensionType" minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

```

[top](#)

Complex Type: **NonOrderedLocations**

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

Sub-types:

- [NonOrderedLocationGroupByList](#) (by extension)
- [NonOrderedLocationGroupByReference](#) (by extension)

Name	NonOrderedLocations
<u>Abstract</u>	yes
Documentation	Multiple (i.e. more than one) physically separate locations which have no specific order.

XML Instance Representation

```

<...>
  <D2LogicalModel:groupOfLocationsExtension>
D2LogicalModel: ExtensionType
</D2LogicalModel:groupOfLocationsExtension> [0..1]
  <D2LogicalModel:nonOrderedLocationsExtension>
D2LogicalModel: ExtensionType
</D2LogicalModel:nonOrderedLocationsExtension> [0..1]
</...>

```

Schema Component Representation

```

<xs:complexType name="NonOrderedLocations" abstract="true">
  <xs:complexContent>
    <xs:extension base="D2LogicalModel:GroupOfLocations">

```

```

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

```

[top](#)

Complex Type: **NumberOfAxlesCharacteristic**

Super-types: None

Sub-types: None

Name NumberOfAxlesCharacteristic
Abstract no
Documentation Number of axles characteristic of a vehicle.

XML Instance Representation

```

<...>
  <D2LogicalModel:comparisonOperator>
    D2LogicalModel:ComparisonOperatorEnum
  </D2LogicalModel:comparisonOperator> [1] ?
  <D2LogicalModel:numberOfAxles> D2LogicalModel:NonNegativeInteger
</D2LogicalModel:numberOfAxles> [1] ?
  <D2LogicalModel:numberOfAxlesCharacteristicExtension>
    D2LogicalModel:_ExtensionType
  </D2LogicalModel:numberOfAxlesCharacteristicExtension> [0..1]
</...>

```

Schema Component Representation

```

<xs:complexType name="NumberOfAxlesCharacteristic">
  <xs:sequence>
    <xs:element name="comparisonOperator"
      type="D2LogicalModel:ComparisonOperatorEnum" minOccurs="1"
      maxOccurs="1"/>
    <xs:element name="numberOfAxles"
      type="D2LogicalModel:NonNegativeInteger" minOccurs="1"
      maxOccurs="1"/>
    <xs:element name="numberOfAxlesCharacteristicExtension"
      type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
  </xs:sequence>
</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: OpeningTimes

Super-types:	None
Sub-types:	None

Name	OpeningTimes
Abstract	no
Documentation	A specification of opening times (e.g. for a parking site, a service facility, an access or the availability for equipment).

XML Instance Representation

```
<...>
  <D2LogicalModel:lastUpdated> D2LogicalModel:DateTime
</D2LogicalModel:lastUpdated> [0..1] ?
  <D2LogicalModel:openAllYear> D2LogicalModel:Boolean
</D2LogicalModel:openAllYear> [0..1] ?
  <D2LogicalModel:available24hours> D2LogicalModel:Boolean
</D2LogicalModel:available24hours> [0..1] ?
  <D2LogicalModel:urlLinkAddress> D2LogicalModel:Url
</D2LogicalModel:urlLinkAddress> [0..1] ?
  <D2LogicalModel:openingTimesUnknown> D2LogicalModel:Boolean
</D2LogicalModel:openingTimesUnknown> [0..1] ?
  <D2LogicalModel:openingTimesNotSpecified> D2LogicalModel:Boolean
</D2LogicalModel:openingTimesNotSpecified> [0..1] ?
  <D2LogicalModel:validity> D2LogicalModel:Validity
</D2LogicalModel:validity> [0..1]
  <D2LogicalModel:openingTimesExtension>
    D2LogicalModel:_ExtensionType
  </D2LogicalModel:openingTimesExtension> [0..1]
</...>
```

```
</D2LogicalModel:openingTimesExtension> [0..1]
</...>
```

Schema Component Representation

```
<xs:complexType name="OpeningTimes">
  <xs:sequence>
    <xs:element name="lastUpdated" type="D2LogicalModel:DateTime"
      minOccurs="0" maxOccurs="1"/>
    <xs:element name="openAllYear" type="D2LogicalModel:Boolean"
      minOccurs="0" maxOccurs="1"/>
    <xs:element name="available24hours"
      type="D2LogicalModel:Boolean" minOccurs="0" maxOccurs="1"/>
    <xs:element name="urlLinkAddress" type="D2LogicalModel:Url"
      minOccurs="0" maxOccurs="1"/>
    <xs:element name="openingTimesUnknown"
      type="D2LogicalModel:Boolean" minOccurs="0" maxOccurs="1"/>
    <xs:element name="openingTimesNotSpecified"
      type="D2LogicalModel:Boolean" minOccurs="0" maxOccurs="1"/>
    <xs:element name="validity" type="D2LogicalModel:Validity"
      minOccurs="0"/>
    <xs:element name="openingTimesExtension"
      type="D2LogicalModel:ExtensionType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
```

[top](#)

Complex Type: OverallPeriod

Super-types: None

Sub-types: None

Name OverallPeriod

Abstract no

Documentation A continuous or discontinuous period of validity defined by overall bounding start and end times and the possible intersection of valid periods (potentially recurring) with the complement of exception periods (also potentially recurring).

XML Instance Representation

```
<...>
  <D2LogicalModel:overallStartTime> D2LogicalModel:DateTime
</D2LogicalModel:overallStartTime> [1] ?
  <D2LogicalModel:overallEndTime> D2LogicalModel:DateTime
</D2LogicalModel:overallEndTime> [0..1] ?
  <D2LogicalModel:validPeriod> D2LogicalModel:Period
</D2LogicalModel:validPeriod> [0..*] ?
  <D2LogicalModel:exceptionPeriod> D2LogicalModel:Period
</D2LogicalModel:exceptionPeriod> [0..*] ?
  <D2LogicalModel:overallPeriodExtension>
    D2LogicalModel:ExtensionType
  </D2LogicalModel:overallPeriodExtension> [0..1]
</...>
```

Schema Component Representation

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

```

<xs:sequence>
  <xs:element name="overallStartTime"
    type="D2LogicalModel:DateTime" minOccurs="1" maxOccurs="1"/>
  <xs:element name="overallEndTime" type="D2LogicalModel:DateTime"
    minOccurs="0" maxOccurs="1"/>
  <xs:element name="validPeriod" type="D2LogicalModel:Period"
    minOccurs="0" maxOccurs="unbounded"/>
  <xs:element name="exceptionPeriod" type="D2LogicalModel:Period"
    minOccurs="0" maxOccurs="unbounded"/>
  <xs:element name="overallPeriodExtension"
    type="D2LogicalModel:ExtensionType" minOccurs="0"/>
</xs:sequence>
</xs:complexType>

```

[top](#)

Complex Type: **ParkingAccess**

Super-types: None

Sub-types: None

Name	ParkingAccess
<u>Abstract</u>	no
Documentation	Describes one entrance or exit (or both) to a parking site.

XML Instance Representation

```

<...
  id="xs:string [1]">
    <D2LogicalModel:accessCategory> D2LogicalModel:AccessCategoryEnum
    </D2LogicalModel:accessCategory> [1..*] ?
    <D2LogicalModel:accessName> D2LogicalModel:MultilingualString
    </D2LogicalModel:accessName> [0..1] ?
    <D2LogicalModel:accessEquipment> D2LogicalModel:AccessEquipmentEnum
    </D2LogicalModel:accessEquipment> [0..*] ?
    <D2LogicalModel:accessibility> D2LogicalModel:AccessibilityEnum
    </D2LogicalModel:accessibility> [0..*] ?
    <D2LogicalModel:photoUrl> D2LogicalModel:Url
    </D2LogicalModel:photoUrl> [0..1] ?
    <D2LogicalModel:accessOnlyAssignedFor>
      D2LogicalModel:ParkingAssignment
    </D2LogicalModel:accessOnlyAssignedFor> [0..1] ?
    <D2LogicalModel:accessAssignedAmongOthers>
      D2LogicalModel:ParkingAssignment
    </D2LogicalModel:accessAssignedAmongOthers> [0..1] ?
    <D2LogicalModel:accessProhibitedFor>
      D2LogicalModel:ParkingAssignment
    </D2LogicalModel:accessProhibitedFor> [0..1] ?
    <D2LogicalModel:primaryRoad> D2LogicalModel:Road
    </D2LogicalModel:primaryRoad> [1..*] ?
    <D2LogicalModel:location> D2LogicalModel:Location
    </D2LogicalModel:location> [1]
    <D2LogicalModel:openingTimes> D2LogicalModel:OpeningTimes
    </D2LogicalModel:openingTimes> [0..1]
    <D2LogicalModel:parkingAccessExtension>
      D2LogicalModel:ExtensionType
    </D2LogicalModel:parkingAccessExtension> [0..1]
  </...>

```

Schema Component Representation

```
<xs:complexType name="ParkingAccess">
  <xs:sequence>
    <xs:element name="accessCategory"
      type="D2LogicalModel:AccessCategoryEnum" minOccurs="1"
      maxOccurs="unbounded"/>
    <xs:element name="accessName"
      type="D2LogicalModel:MultilingualString" minOccurs="0"
      maxOccurs="1"/>
    <xs:element name="accessEquipment"
      type="D2LogicalModel:AccessEquipmentEnum" minOccurs="0"
      maxOccurs="unbounded"/>
    <xs:element name="accessibility"
      type="D2LogicalModel:AccessibilityEnum" minOccurs="0"
      maxOccurs="unbounded"/>
    <xs:element name="photoUrl" type="D2LogicalModel:Url"
      minOccurs="0" maxOccurs="1"/>
    <xs:element name="accessOnlyAssignedFor"
      type="D2LogicalModel:ParkingAssignment" minOccurs="0"/>
    <xs:element name="accessAssignedAmongOthers"
      type="D2LogicalModel:ParkingAssignment" minOccurs="0"/>
    <xs:element name="accessProhibitedFor"
      type="D2LogicalModel:ParkingAssignment" minOccurs="0"/>
    <xs:element name="primaryRoad" type="D2LogicalModel:Road"
      maxOccurs="unbounded"/>
    <xs:element name="location" type="D2LogicalModel:Location"/>
    <xs:element name="openingTimes"
      type="D2LogicalModel:OpeningTimes" minOccurs="0"/>
    <xs:element name="parkingAccessExtension"
      type="D2LogicalModel:ExtensionType" minOccurs="0"/>
  </xs:sequence>
  <xs:attribute name="id" type="xs:string" use="required"/>
</xs:complexType>
```

[top](#)

Complex Type: **ParkingAssignment**

Super-types: None

Sub-types: None

Name ParkingAssignment

Abstract no

Documentation One set of prohibited/only allowed/convenient assignment for parking space(s), parking site(s) or an access. Same kind of data forms a union (e.g. lorries OR buses), different kind of data forms an intersection (e.g. residents AND long-term).

XML Instance Representation

```
<...>
  <D2LogicalModel:applicableForUser> D2LogicalModel:UserTypeEnum
</D2LogicalModel:applicableForUser> [0..*] ?
  <D2LogicalModel:parkingDuration> D2LogicalModel:ParkingDurationEnum
</D2LogicalModel:parkingDuration> [0..*] ?
  <D2LogicalModel:vehicleCharacteristics>
D2LogicalModel:VehicleCharacteristics
</D2LogicalModel:vehicleCharacteristics> [0..*]
  <D2LogicalModel:hazardousMaterials>
```

```

D2LogicalModel:HazardousMaterials
</D2LogicalModel:hazardousMaterials> [0..*] ?
<D2LogicalModel:timePeriodByHour> D2LogicalModel:TimePeriodByHour
</D2LogicalModel:timePeriodByHour> [0..*] ?
<D2LogicalModel:parkingPermit> D2LogicalModel:ParkingPermit
</D2LogicalModel:parkingPermit> [0..*]
<D2LogicalModel:parkingAssignmentExtension>
D2LogicalModel:ExtensionType
</D2LogicalModel:parkingAssignmentExtension> [0..1]
</...>

```

Schema Component Representation

```

<xs:complexType name="ParkingAssignment">
  <xs:sequence>
    <xs:element name="applicableForUser"
      type="D2LogicalModel:UserTypeEnum" minOccurs="0"
      maxOccurs="unbounded"/>
    <xs:element name="parkingDuration"
      type="D2LogicalModel:ParkingDurationEnum" minOccurs="0"
      maxOccurs="unbounded"/>
    <xs:element name="vehicleCharacteristics"
      type="D2LogicalModel:VehicleCharacteristics" minOccurs="0"
      maxOccurs="unbounded"/>
    <xs:element name="hazardousMaterials"
      type="D2LogicalModel:HazardousMaterials" minOccurs="0"
      maxOccurs="unbounded"/>
    <xs:element name="timePeriodByHour"
      type="D2LogicalModel:TimePeriodByHour" minOccurs="0"
      maxOccurs="unbounded"/>
    <xs:element name="parkingPermit"
      type="D2LogicalModel:ParkingPermit" minOccurs="0"
      maxOccurs="unbounded"/>
    <xs:element name="parkingAssignmentExtension"
      type="D2LogicalModel:ExtensionType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>

```

[top](#)

Complex Type: **ParkingEquipmentOrServiceFacility**

Super-types: None

Sub-types:

- [Equipment](#) (by extension)
- [ServiceFacility](#) (by extension)

Name	ParkingEquipmentOrServiceFacility
Abstract	yes
Documentation	One type of equipment or additional service facility that is available at the parking site, parking space or group of parking spaces.

XML Instance Representation

```

<...>
  <D2LogicalModel:equipmentOrServiceFacilityIdentifier>
D2LogicalModel:String
  </D2LogicalModel:equipmentOrServiceFacilityIdentifier> [0..*] ?
  <D2LogicalModel:availability> D2LogicalModel:AvailabilityEnum

```

```

</D2LogicalModel:availability> [0..1] ?
<D2LogicalModel:numberOfEquipmentOrServiceFacility>
D2LogicalModel:NonNegativeInteger
</D2LogicalModel:numberOfEquipmentOrServiceFacility> [0..1] ?
<D2LogicalModel:additionalDescription>
D2LogicalModel:MultilingualString
</D2LogicalModel:additionalDescription> [0..1] ?
<D2LogicalModel:otherEquipmentOrServiceFacility>
D2LogicalModel:MultilingualString
</D2LogicalModel:otherEquipmentOrServiceFacility> [0..1] ?
<D2LogicalModel:accessibility> D2LogicalModel:AccessibilityEnum
</D2LogicalModel:accessibility> [0..*] ?
<D2LogicalModel:nameOrBrand> D2LogicalModel:MultilingualString
</D2LogicalModel:nameOrBrand> [0..1] ?
<D2LogicalModel:comment> D2LogicalModel:MultilingualString
</D2LogicalModel:comment> [0..1] ?
<D2LogicalModel:photoUrl> D2LogicalModel:Url
</D2LogicalModel:photoUrl> [0..1] ?
<D2LogicalModel:applicableForUser> D2LogicalModel:UserTypeEnum
</D2LogicalModel:applicableForUser> [0..*] ?
<D2LogicalModel:availabilityAndOpeningTimes>
D2LogicalModel:OpeningTimes
</D2LogicalModel:availabilityAndOpeningTimes> [0..1] ?
<D2LogicalModel:tariffsAndPayment> D2LogicalModel:TariffsAndPayment
</D2LogicalModel:tariffsAndPayment> [0..1]
<D2LogicalModel:groupOfLocations> D2LogicalModel:GroupOfLocations
</D2LogicalModel:groupOfLocations> [0..1]
<D2LogicalModel:applicableForVehicles>
D2LogicalModel:VehicleCharacteristics
</D2LogicalModel:applicableForVehicles> [0..*]
<D2LogicalModel:parkingEquipmentOrServiceFacilityExtension>
D2LogicalModel:ExtensionType
</D2LogicalModel:parkingEquipmentOrServiceFacilityExtension> [0..1]
</...>

```

Schema Component Representation

```

<xs:complexType name="ParkingEquipmentOrServiceFacility"
abstract="true">
  <xs:sequence>
    <xs:element name="equipmentOrServiceFacilityIdentifier"
      type="D2LogicalModel:String" minOccurs="0"
      maxOccurs="unbounded"/>
    <xs:element name="availability"
      type="D2LogicalModel:AvailabilityEnum" minOccurs="0"
      maxOccurs="1"/>
    <xs:element name="numberOfEquipmentOrServiceFacility"
      type="D2LogicalModel:NonNegativeInteger" minOccurs="0"
      maxOccurs="1"/>
    <xs:element name="additionalDescription"
      type="D2LogicalModel:MultilingualString" minOccurs="0"
      maxOccurs="1"/>
    <xs:element name="otherEquipmentOrServiceFacility"
      type="D2LogicalModel:MultilingualString" minOccurs="0"
      maxOccurs="1"/>
    <xs:element name="accessibility"
      type="D2LogicalModel:AccessibilityEnum" minOccurs="0"
      maxOccurs="unbounded"/>
    <xs:element name="nameOrBrand"
      type="D2LogicalModel:MultilingualString" minOccurs="0"
      maxOccurs="1"/>
    <xs:element name="comment"
      type="D2LogicalModel:MultilingualString" minOccurs="0"

```

```

maxOccurs="1"/>
<xs:element name="photoUrl" type="D2LogicalModel:Url"
minOccurs="0" maxOccurs="1"/>
<xs:element name="applicableForUser"
type="D2LogicalModel:UserTypeEnum" minOccurs="0"
maxOccurs="unbounded"/>
<xs:element name="availabilityAndOpeningTimes"
type="D2LogicalModel:OpeningTimes" minOccurs="0"/>
<xs:element name="tariffsAndPayment"
type="D2LogicalModel:TariffsAndPayment" minOccurs="0"/>
<xs:element name="groupOfLocations"
type="D2LogicalModel:GroupOfLocations" minOccurs="0"/>
<xs:element name="applicableForVehicles"
type="D2LogicalModel:VehicleCharacteristics" minOccurs="0"
maxOccurs="unbounded"/>
<xs:element name="parkingEquipmentOrServiceFacilityExtension"
type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
</xs:sequence>
</xs:complexType>

```

[top](#)

Complex Type: **ParkingPermit**

Super-types: None

Sub-types: None

Name	ParkingPermit
Abstract	no
Documentation	A permission for parking.

XML Instance Representation

```

<...>
<D2LogicalModel:parkingPermitType> D2LogicalModel:PermitTypeEnum
</D2LogicalModel:parkingPermitType> [1] ?
<D2LogicalModel:parkingPermitScheme> D2LogicalModel:String
</D2LogicalModel:parkingPermitScheme> [0..1] ?
<D2LogicalModel:parkingPermitIdentifier> D2LogicalModel:String
</D2LogicalModel:parkingPermitIdentifier> [0..1] ?
<D2LogicalModel:parkingPermitExtension>
D2LogicalModel:_ExtensionType
</D2LogicalModel:parkingPermitExtension> [0..1]
</...>

```

Schema Component Representation

```

<xs:complexType name="ParkingPermit">
  <xs:sequence>
    <xs:element name="parkingPermitType"
      type="D2LogicalModel:PermitTypeEnum" minOccurs="1"
      maxOccurs="1"/>
    <xs:element name="parkingPermitScheme"
      type="D2LogicalModel:String" minOccurs="0" maxOccurs="1"/>
    <xs:element name="parkingPermitIdentifier"
      type="D2LogicalModel:String" minOccurs="0" maxOccurs="1"/>
    <xs:element name="parkingPermitExtension"
      type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
  </xs:sequence>

```

Complex Type: **ParkingRecord**

Super-types: None

Sub-types:

- [GroupOfParkingSites](#) (by extension)
- [ParkingSite](#) (by extension)
 - [InterUrbanParkingSite](#) (by extension)
 - [SpecialLocationParkingSite](#) (by extension)
 - [UrbanParkingSite](#) (by extension)

Name ParkingRecord

Abstract yes

Documentation A container for static parking information. Must be specialised as a parking site or as a group of parking sites.

XML Instance Representation

```
<...  
  id="xs:string [1]"  
  version="xs:string [1]">  
    <D2LogicalModel:parkingName> D2LogicalModel:MultilingualString  
    </D2LogicalModel:parkingName> [1] ?  
    <D2LogicalModel:parkingAlias> D2LogicalModel:MultilingualString  
    </D2LogicalModel:parkingAlias> [0..*] ?  
    <D2LogicalModel:parkingDescription>  
      D2LogicalModel:MultilingualString  
    </D2LogicalModel:parkingDescription> [0..1] ?  
    <D2LogicalModel:parkingRecordVersionTime> D2LogicalModel:DateTime  
    </D2LogicalModel:parkingRecordVersionTime> [1] ?  
    <D2LogicalModel:parkingNumberOfSpaces>  
      D2LogicalModel:NonNegativeInteger  
    </D2LogicalModel:parkingNumberOfSpaces> [0..1] ?  
    <D2LogicalModel:parkingPrincipalNumberOfSpaces>  
      D2LogicalModel:NonNegativeInteger  
    </D2LogicalModel:parkingPrincipalNumberOfSpaces> [0..1] ?  
    <D2LogicalModel:maximumParkingDuration> D2LogicalModel:Seconds  
    </D2LogicalModel:maximumParkingDuration> [0..1] ?  
    <D2LogicalModel:photoUrl> D2LogicalModel:Url  
    </D2LogicalModel:photoUrl> [0..1] ?  
    <D2LogicalModel:urlLinkAddress> D2LogicalModel:Url  
    </D2LogicalModel:urlLinkAddress> [0..1] ?  
    <D2LogicalModel:parkingOccupanyDetectionType>  
      D2LogicalModel:OccupancyDetectionTypeEnum  
    </D2LogicalModel:parkingOccupanyDetectionType> [0..*] ?  
    <D2LogicalModel:emergencyContact> D2LogicalModel:Contact  
    </D2LogicalModel:emergencyContact> [0..*] ?  
    <D2LogicalModel:owner> D2LogicalModel:Contact  
    </D2LogicalModel:owner> [0..*] ?  
    <D2LogicalModel:responisbleAuthority> D2LogicalModel:Contact  
    </D2LogicalModel:responisbleAuthority> [0..*] ?  
    <D2LogicalModel:securityService> D2LogicalModel:Contact  
    </D2LogicalModel:securityService> [0..*] ?  
    <D2LogicalModel:operator> D2LogicalModel:Contact  
    </D2LogicalModel:operator> [1..*] ?  
  </...>
```

```

<D2LogicalModel:servicePartner> D2LogicalModel:Contact
</D2LogicalModel:servicePartner> [0..*] ?
<D2LogicalModel:parkingVMS> D2LogicalModel:ParkingVMS
</D2LogicalModel:parkingVMS> [0..*]
<D2LogicalModel:parkingLocation> D2LogicalModel:GroupOfLocations
</D2LogicalModel:parkingLocation> [1] ?
<D2LogicalModel:parkingRoute> D2LogicalModel:ParkingRoute
</D2LogicalModel:parkingRoute> [0..*]
<D2LogicalModel:parkingColour> D2LogicalModel:RGBColour
</D2LogicalModel:parkingColour> [0..1] ?
<D2LogicalModel:onlyAssignedParking>
D2LogicalModel:ParkingAssignment
</D2LogicalModel:onlyAssignedParking> [0..1] ?
<D2LogicalModel:assignedParkingAmongOthers>
D2LogicalModel:ParkingAssignment
</D2LogicalModel:assignedParkingAmongOthers> [0..1] ?
<D2LogicalModel:prohibitedParking> D2LogicalModel:ParkingAssignment
</D2LogicalModel:prohibitedParking> [0..1] ?
<D2LogicalModel:tariffsAndPayment> D2LogicalModel:TariffsAndPayment
</D2LogicalModel:tariffsAndPayment> [0..1]
<D2LogicalModel:parkingEquipmentOrServiceFacility>
D2LogicalModel: ParkingRecordEquipmentOrServiceFacilityIndexParkingEquipmen
</D2LogicalModel:parkingEquipmentOrServiceFacility> [0..*]
<D2LogicalModel:parkingSpace> D2LogicalModel: ParkingSpace
</D2LogicalModel:parkingSpace> [0..*] ?
<D2LogicalModel:groupOfParkingSpaces>
D2LogicalModel: GroupOfParkingSpaces
</D2LogicalModel:groupOfParkingSpaces> [0..*] ?
<D2LogicalModel:parkingThresholds> D2LogicalModel:ParkingThresholds
</D2LogicalModel:parkingThresholds> [0..1]
<D2LogicalModel:permitsAndProhibitions>
D2LogicalModel:PermitsAndProhibitions
</D2LogicalModel:permitsAndProhibitions> [0..*]
<D2LogicalModel:emergencyAssemblyPoint>
D2LogicalModel:GroupOfLocations
</D2LogicalModel:emergencyAssemblyPoint> [0..1] ?
<D2LogicalModel:entireArea> D2LogicalModel:Area
</D2LogicalModel:entireArea> [0..1] ?
<D2LogicalModel:parkingRecordDimension> D2LogicalModel:Dimension
</D2LogicalModel:parkingRecordDimension> [0..1] ?
<D2LogicalModel:parkingRecordExtension>
D2LogicalModel: ExtensionType
</D2LogicalModel:parkingRecordExtension> [0..1]
</...>

```

Schema Component Representation

```

<xs:complexType name="ParkingRecord" abstract="true">
  <xs:sequence>
    <xs:element name="parkingName"
      type="D2LogicalModel:MultilingualString" minOccurs="1"
      maxOccurs="1"/>
    <xs:element name="parkingAlias"
      type="D2LogicalModel:MultilingualString" minOccurs="0"
      maxOccurs="unbounded"/>
    <xs:element name="parkingDescription"
      type="D2LogicalModel:MultilingualString" minOccurs="0"
      maxOccurs="1"/>
    <xs:element name="parkingRecordVersionTime"
      type="D2LogicalModel:DateTime" minOccurs="1" maxOccurs="1"/>
    <xs:element name="parkingNumberOfSpaces"
      type="D2LogicalModel:NonNegativeInteger" minOccurs="0"
      maxOccurs="1"/>
  
```

```

<xs:element name="parkingPrincipalNumberOfSpaces"
type="D2LogicalModel:NonNegativeInteger" minOccurs="0"
maxOccurs="1"/>
<xs:element name="maximumParkingDuration"
type="D2LogicalModel:Seconds" minOccurs="0" maxOccurs="1"/>
<xs:element name="photoUrl" type="D2LogicalModel:Url"
minOccurs="0" maxOccurs="1"/>
<xs:element name="urlLinkAddress" type="D2LogicalModel:Url"
minOccurs="0" maxOccurs="1"/>
<xs:element name="parkingOccupanyDetectionType"
type="D2LogicalModel:OccupancyDetectionTypeEnum" minOccurs="0"
maxOccurs="unbounded"/>
<xs:element name="emergencyContact"
type="D2LogicalModel:Contact" minOccurs="0"
maxOccurs="unbounded"/>
<xs:element name="owner" type="D2LogicalModel:Contact"
minOccurs="0" maxOccurs="unbounded"/>
<xs:element name="responisbleAuthority"
type="D2LogicalModel:Contact" minOccurs="0"
maxOccurs="unbounded"/>
<xs:element name="securityService" type="D2LogicalModel:Contact"
minOccurs="0" maxOccurs="unbounded"/>
<xs:element name="operator" type="D2LogicalModel:Contact"
maxOccurs="unbounded"/>
<xs:element name="servicePartner" type="D2LogicalModel:Contact"
minOccurs="0" maxOccurs="unbounded"/>
<xs:element name="parkingVMS" type="D2LogicalModel:ParkingVMS"
minOccurs="0" maxOccurs="unbounded"/>
<xs:element name="parkingLocation"
type="D2LogicalModel:GroupOfLocations"/>
<xs:element name="parkingRoute"
type="D2LogicalModel:ParkingRoute" minOccurs="0"
maxOccurs="unbounded"/>
<xs:element name="parkingColour" type="D2LogicalModel:RGBColour"
minOccurs="0"/>
<xs:element name="onlyAssignedParking"
type="D2LogicalModel:ParkingAssignment" minOccurs="0"/>
<xs:element name="assignedParkingAmongOthers"
type="D2LogicalModel:ParkingAssignment" minOccurs="0"/>
<xs:element name="prohibitedParking"
type="D2LogicalModel:ParkingAssignment" minOccurs="0"/>
<xs:element name="tariffsAndPayment"
type="D2LogicalModel:TariffsAndPayment" minOccurs="0"/>
<xs:element name="parkingEquipmentOrServiceFacility"
type="D2LogicalModel: ParkingRecordEquipmentOrServiceFacilityIndexParking"
minOccurs="0" maxOccurs="unbounded"/>
<xs:element name="parkingSpace"
type="D2LogicalModel: ParkingSpace" minOccurs="0"
maxOccurs="unbounded"/>
<xs:element name="groupOfParkingSpaces"
type="D2LogicalModel: GroupOfParkingSpaces" minOccurs="0"
maxOccurs="unbounded"/>
<xs:element name="parkingThresholds"
type="D2LogicalModel: ParkingThresholds" minOccurs="0"/>
<xs:element name="permitsAndProhibitions"
type="D2LogicalModel:PermitsAndProhibitions" minOccurs="0"
maxOccurs="unbounded"/>
<xs:element name="emergencyAssemblyPoint"
type="D2LogicalModel:GroupOfLocations" minOccurs="0"/>
<xs:element name="entireArea" type="D2LogicalModel:Area"
minOccurs="0"/>
<xs:element name="parkingRecordDimension"
type="D2LogicalModel:Dimension" minOccurs="0"/>
<xs:element name="parkingRecordExtension"

```

```

    type="D2LogicalModel: _ExtensionType" minOccurs="0"/>
</xs:sequence>
<xs:attribute name="id" type="xs:string" use="required"/>
<xs:attribute name="version" type="xs:string" use="required"/>
</xs:complexType>

```

[top](#)

Complex Type: **ParkingRoute**

Super-types: None

Sub-types:

- [ParkingRouteByReference](#) (by extension)
- [ParkingRouteDetails](#) (by extension)

Name ParkingRoute

Abstract yes

Documentation A parking route, defined by ParkingRouteDetails or by a reference.

XML Instance Representation

```

<...>
  <D2LogicalModel:parkingRouteColour> D2LogicalModel:RGBColour
</D2LogicalModel:parkingRouteColour> [0..1] ?
  <D2LogicalModel:parkingRouteExtension>
D2LogicalModel: _ExtensionType
</D2LogicalModel:parkingRouteExtension> [0..1]
</...>

```

Schema Component Representation

```

<xs:complexType name="ParkingRoute" abstract="true">
  <xs:sequence>
    <xs:element name="parkingRouteColour"
      type="D2LogicalModel:RGBColour" minOccurs="0"/>
    <xs:element name="parkingRouteExtension"
      type="D2LogicalModel: _ExtensionType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>

```

[top](#)

Complex Type: **ParkingRouteByReference**

Super-types: [ParkingRoute](#) < **ParkingRouteByReference** (by extension)

Sub-types: None

Name ParkingRouteByReference

Abstract no

Documentation A route defined by a reference to an earlier specified route.

XML Instance Representation

```

<...>

```

```

<D2LogicalModel:parkingRouteColour> D2LogicalModel:RGBColour
</D2LogicalModel:parkingRouteColour> [0..1] ?
<D2LogicalModel:parkingRouteExtension>
D2LogicalModel: ExtensionType
</D2LogicalModel:parkingRouteExtension> [0..1]
<D2LogicalModel:parkingRouteReference>
D2LogicalModel: ParkingRouteDetailsVersionedReference
</D2LogicalModel:parkingRouteReference> [1] ?
<D2LogicalModel:parkingRouteByReferenceExtension>
D2LogicalModel: ExtensionType
</D2LogicalModel:parkingRouteByReferenceExtension> [0..1]
</...>

```

Schema Component Representation

```

<xs:complexType name="ParkingRouteByReference">
  <xs:complexContent>
    <xs:extension base="D2LogicalModel: ParkingRoute">
      <xs:sequence>
        <xs:element name="parkingRouteReference"
          type="D2LogicalModel: ParkingRouteDetailsVersionedReference"
          minOccurs="1" maxOccurs="1"/>
        <xs:element name="parkingRouteByReferenceExtension"
          type="D2LogicalModel: ExtensionType" minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

```

[top](#)

Complex Type: **ParkingRouteDetails**

Super-types: [ParkingRoute](#) < **ParkingRouteDetails** (by extension)

Sub-types: None

Name	ParkingRouteDetails
Abstract	no
Documentation	Urban context: Defining parking routes leading to the parking site. Truck parking context: Can be used to define a dynamic route management.

XML Instance Representation

```

<...
  id="xs:string [1]"
  version="xs:string [1]">
    <D2LogicalModel:parkingRouteColour> D2LogicalModel:RGBColour
    </D2LogicalModel:parkingRouteColour> [0..1] ?
    <D2LogicalModel:parkingRouteExtension>
      D2LogicalModel: ExtensionType
    </D2LogicalModel:parkingRouteExtension> [0..1]
    <D2LogicalModel:parkingRouteName> D2LogicalModel:MultilingualString
    </D2LogicalModel:parkingRouteName> [0..1] ?
    <D2LogicalModel:parkingRouteType>
      D2LogicalModel: ParkingRouteTypeEnum
    </D2LogicalModel:parkingRouteType> [0..1] ?
    <D2LogicalModel:dynamicRouteManagement> D2LogicalModel:Boolean
    </D2LogicalModel:dynamicRouteManagement> [0..1] ?
  </...>

```

```

<D2LogicalModel:parkingRouteIconIndex> D2LogicalModel:String
</D2LogicalModel:parkingRouteIconIndex> [0..1] ?
<D2LogicalModel:parkingRouteDirection> D2LogicalModel:DirectionEnum
</D2LogicalModel:parkingRouteDirection> [0..1] ?
<D2LogicalModel:parkingRouteDirection2>
D2LogicalModel:ParkingRouteDirectionEnum
</D2LogicalModel:parkingRouteDirection2> [0..1] ?
<D2LogicalModel:groupOfLocations> D2LogicalModel:GroupOfLocations
</D2LogicalModel:groupOfLocations> [0..1]
<D2LogicalModel:parkingRouteDetailsExtension>
D2LogicalModel:ExtensionType
</D2LogicalModel:parkingRouteDetailsExtension> [0..1]
</...>

```

Schema Component Representation

```

<xs:complexType name="ParkingRouteDetails">
  <xs:complexContent>
    <xs:extension base="D2LogicalModel:ParkingRoute">
      <xs:sequence>
        <xs:element name="parkingRouteName"
          type="D2LogicalModel:MultilingualString" minOccurs="0"
          maxOccurs="1"/>
        <xs:element name="parkingRouteType"
          type="D2LogicalModel:ParkingRouteTypeEnum" minOccurs="0"
          maxOccurs="1"/>
        <xs:element name="dynamicRouteManagement"
          type="D2LogicalModel:Boolean" minOccurs="0" maxOccurs="1"/>
        <xs:element name="parkingRouteIconIndex"
          type="D2LogicalModel:String" minOccurs="0" maxOccurs="1"/>
        <xs:element name="parkingRouteDirection"
          type="D2LogicalModel:DirectionEnum" minOccurs="0"
          maxOccurs="1"/>
        <xs:element name="parkingRouteDirection2"
          type="D2LogicalModel:ParkingRouteDirectionEnum"
          minOccurs="0" maxOccurs="1"/>
        <xs:element name="groupOfLocations"
          type="D2LogicalModel:GroupOfLocations" minOccurs="0"/>
        <xs:element name="parkingRouteDetailsExtension"
          type="D2LogicalModel:ExtensionType" minOccurs="0"/>
      </xs:sequence>
      <xs:attribute name="id" type="xs:string" use="required"/>
      <xs:attribute name="version" type="xs:string" use="required"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

```

[top](#)

Complex Type: **ParkingSite**

Super-types: [ParkingRecord](#) < **ParkingSite** (by extension)

Sub-types:

- [InterUrbanParkingSite](#) (by extension)
- [SpecialLocationParkingSite](#) (by extension)
- [UrbanParkingSite](#) (by extension)

Name ParkingSite

Abstract yes

Documentation

A record containing static details of a parking site. Must be specialised as an 'Urban-' or 'InterUrbanParkingSite' or a 'SpecialLocationParkingSite'.

XML Instance Representation

```
<...
  id="xs:string [1]"
  version="xs:string [1]">
    <D2LogicalModel:parkingName> D2LogicalModel:MultilingualString
    </D2LogicalModel:parkingName> [1] ?
    <D2LogicalModel:parkingAlias> D2LogicalModel:MultilingualString
    </D2LogicalModel:parkingAlias> [0..*] ?
    <D2LogicalModel:parkingDescription>
      D2LogicalModel:MultilingualString
    </D2LogicalModel:parkingDescription> [0..1] ?
    <D2LogicalModel:parkingRecordVersionTime> D2LogicalModel:DateTime
    </D2LogicalModel:parkingRecordVersionTime> [1] ?
    <D2LogicalModel:parkingNumberOfSpaces>
      D2LogicalModel:NonNegativeInteger
    </D2LogicalModel:parkingNumberOfSpaces> [0..1] ?
    <D2LogicalModel:parkingPrincipalNumberOfSpaces>
      D2LogicalModel:NonNegativeInteger
    </D2LogicalModel:parkingPrincipalNumberOfSpaces> [0..1] ?
    <D2LogicalModel:maximumParkingDuration> D2LogicalModel:Seconds
    </D2LogicalModel:maximumParkingDuration> [0..1] ?
    <D2LogicalModel:photoUrl> D2LogicalModel:Url
    </D2LogicalModel:photoUrl> [0..1] ?
    <D2LogicalModel:urlLinkAddress> D2LogicalModel:Url
    </D2LogicalModel:urlLinkAddress> [0..1] ?
    <D2LogicalModel:parkingOccupancyDetectionType>
      D2LogicalModel:OccupancyDetectionTypeEnum
    </D2LogicalModel:parkingOccupancyDetectionType> [0..*] ?
    <D2LogicalModel:emergencyContact> D2LogicalModel:Contact
    </D2LogicalModel:emergencyContact> [0..*] ?
    <D2LogicalModel:owner> D2LogicalModel:Contact
    </D2LogicalModel:owner> [0..*] ?
    <D2LogicalModel:responsibleAuthority> D2LogicalModel:Contact
    </D2LogicalModel:responsibleAuthority> [0..*] ?
    <D2LogicalModel:securityService> D2LogicalModel:Contact
    </D2LogicalModel:securityService> [0..*] ?
    <D2LogicalModel:operator> D2LogicalModel:Contact
    </D2LogicalModel:operator> [1..*] ?
    <D2LogicalModel:servicePartner> D2LogicalModel:Contact
    </D2LogicalModel:servicePartner> [0..*] ?
    <D2LogicalModel:parkingVMS> D2LogicalModel:ParkingVMS
    </D2LogicalModel:parkingVMS> [0..*]
    <D2LogicalModel:parkingLocation> D2LogicalModel:GroupOfLocations
    </D2LogicalModel:parkingLocation> [1] ?
    <D2LogicalModel:parkingRoute> D2LogicalModel:ParkingRoute
    </D2LogicalModel:parkingRoute> [0..*]
    <D2LogicalModel:parkingColour> D2LogicalModel:RGBColour
    </D2LogicalModel:parkingColour> [0..1] ?
    <D2LogicalModel:onlyAssignedParking>
      D2LogicalModel:ParkingAssignment
    </D2LogicalModel:onlyAssignedParking> [0..1] ?
    <D2LogicalModel:assignedParkingAmongOthers>
      D2LogicalModel:ParkingAssignment
    </D2LogicalModel:assignedParkingAmongOthers> [0..1] ?
    <D2LogicalModel:prohibitedParking> D2LogicalModel:ParkingAssignment
    </D2LogicalModel:prohibitedParking> [0..1] ?
    <D2LogicalModel:tariffsAndPayment> D2LogicalModel:TariffsAndPayment
```

```

</D2LogicalModel:tariffsAndPayment> [0..1]
<D2LogicalModel:parkingEquipmentOrServiceFacility>
D2LogicalModel: ParkingRecordEquipmentOrServiceFacilityIndexParkingEquipmen
</D2LogicalModel:parkingEquipmentOrServiceFacility> [0..*]
<D2LogicalModel:parkingSpace> D2LogicalModel: ParkingSpace
</D2LogicalModel:parkingSpace> [0..*] ?
<D2LogicalModel:groupOfParkingSpaces>
D2LogicalModel: GroupOfParkingSpaces
</D2LogicalModel:groupOfParkingSpaces> [0..*] ?
<D2LogicalModel:parkingThresholds> D2LogicalModel: ParkingThresholds
</D2LogicalModel:parkingThresholds> [0..1]
<D2LogicalModel:permitsAndProhibitions>
D2LogicalModel: PermitsAndProhibitions
</D2LogicalModel:permitsAndProhibitions> [0..*]
<D2LogicalModel:emergencyAssemblyPoint>
D2LogicalModel: GroupOfLocations
</D2LogicalModel:emergencyAssemblyPoint> [0..1] ?
<D2LogicalModel:entireArea> D2LogicalModel: Area
</D2LogicalModel:entireArea> [0..1] ?
<D2LogicalModel:parkingRecordDimension> D2LogicalModel: Dimension
</D2LogicalModel:parkingRecordDimension> [0..1] ?
<D2LogicalModel:parkingRecordExtension>
D2LogicalModel: ExtensionType
</D2LogicalModel:parkingRecordExtension> [0..1]
<D2LogicalModel:parkingReservation>
D2LogicalModel: ReservationTypeEnum
</D2LogicalModel:parkingReservation> [0..1] ?
<D2LogicalModel:parkingLayout> D2LogicalModel: ParkingLayoutEnum
</D2LogicalModel:parkingLayout> [0..*] ?
<D2LogicalModel:highestFloor> D2LogicalModel: Integer
</D2LogicalModel:highestFloor> [0..1] ?
<D2LogicalModel:lowestFloor> D2LogicalModel: Integer
</D2LogicalModel:lowestFloor> [0..1] ?
<D2LogicalModel:temporaryParking> D2LogicalModel: Boolean
</D2LogicalModel:temporaryParking> [0..1] ?
<D2LogicalModel:parkingSiteAddress> D2LogicalModel: Contact
</D2LogicalModel:parkingSiteAddress> [1..*] ?
<D2LogicalModel:reservationService> D2LogicalModel: Contact
</D2LogicalModel:reservationService> [0..*] ?
<D2LogicalModel:parkingUsageScenario>
D2LogicalModel: ParkingSiteScenarioIndexParkingUsageScenario
</D2LogicalModel:parkingUsageScenario> [0..*]
<D2LogicalModel:openingTimes> D2LogicalModel: OpeningTimes
</D2LogicalModel:openingTimes> [0..1]
<D2LogicalModel:parkingAccess> D2LogicalModel: ParkingAccess
</D2LogicalModel:parkingAccess> [1..*] ?
<D2LogicalModel:parkingStandardsAndSecurity>
D2LogicalModel: ParkingStandardsAndSecurity
</D2LogicalModel:parkingStandardsAndSecurity> [1]
<D2LogicalModel:parkingSiteExtension> D2LogicalModel: ExtensionType
</D2LogicalModel:parkingSiteExtension> [0..1]
</...>

```

Schema Component Representation

```

<xs:complexType name="ParkingSite" abstract="true">
  <xs:complexContent>
    <xs:extension base="D2LogicalModel: ParkingRecord">
      <xs:sequence>
        <xs:element name="parkingReservation"
          type="D2LogicalModel: ReservationTypeEnum" minOccurs="0"
          maxOccurs="1"/>
        <xs:element name="parkingLayout"

```

```

type="D2LogicalModel:ParkingLayoutEnum" minOccurs="0"
maxOccurs="unbounded"/>
<xs:element name="highestFloor"
type="D2LogicalModel:Integer" minOccurs="0" maxOccurs="1"/>
<xs:element name="lowestFloor"
type="D2LogicalModel:Integer" minOccurs="0" maxOccurs="1"/>
<xs:element name="temporaryParking"
type="D2LogicalModel:Boolean" minOccurs="0" maxOccurs="1"/>
<xs:element name="parkingSiteAddress"
type="D2LogicalModel:Contact" maxOccurs="unbounded"/>
<xs:element name="reservationService"
type="D2LogicalModel:Contact" minOccurs="0"
maxOccurs="unbounded"/>
<xs:element name="parkingUsageScenario"
type="D2LogicalModel:_ParkingSiteScenarioIndexParkingUsageScenario"
minOccurs="0" maxOccurs="unbounded"/>
<xs:element name="openingTimes"
type="D2LogicalModel:OpeningTimes" minOccurs="0"/>
<xs:element name="parkingAccess"
type="D2LogicalModel:ParkingAccess" maxOccurs="unbounded"/>
<xs:element name="parkingStandardsAndSecurity"
type="D2LogicalModel:ParkingStandardsAndSecurity"/>
<xs:element name="parkingSiteExtension"
type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
</xs:sequence>
</xs:extension>
</xs:complexContent>
</xs:complexType>

```

[top](#)

Complex Type: **ParkingSpace**

Super-types: [ParkingSpaceBasics](#) < **ParkingSpace** (by extension)

Sub-types: None

Name ParkingSpace

Abstract no

Documentation A single parking space. It is possible to define the same parking space more than once with different properties, e.g. when there is a different parking assignment for different times.

XML Instance Representation

```

<...>
  <D2LogicalModel:parkingSpaceOrGroupIdentifier>
    D2LogicalModel:MultilingualString
  </D2LogicalModel:parkingSpaceOrGroupIdentifier> [0..1] ?
  <D2LogicalModel:parkingFloorOrLevel> D2LogicalModel:Integer
  </D2LogicalModel:parkingFloorOrLevel> [0..1] ?
  <D2LogicalModel:accessibility> D2LogicalModel:AccessibilityEnum
  </D2LogicalModel:accessibility> [0..*] ?
  <D2LogicalModel:parkingSpaceAccessibility>
    D2LogicalModel:ParkingSpaceAccessibilityEnum
  </D2LogicalModel:parkingSpaceAccessibility> [0..*] ?
  <D2LogicalModel:parkingSpacePhysics>
    D2LogicalModel:ParkingSpacePhysicsEnum
  </D2LogicalModel:parkingSpacePhysics> [0..2] ?
  <D2LogicalModel:parkingMode> D2LogicalModel:ParkingModeEnum

```

```

</D2LogicalModel:parkingMode> [0..1] ?
<D2LogicalModel:parkingReservation>
D2LogicalModel:ReservationTypeEnum
</D2LogicalModel:parkingReservation> [0..1] ?
<D2LogicalModel:maximumParkingDuration> D2LogicalModel:Seconds
</D2LogicalModel:maximumParkingDuration> [0..1] ?
<D2LogicalModel:distanceFromPrimaryRoad>
D2LogicalModel:MetresAsNonNegativeInteger
</D2LogicalModel:distanceFromPrimaryRoad> [0..1] ?
<D2LogicalModel:parkingOccupanyDetectionType>
D2LogicalModel:OccupancyDetectionTypeEnum
</D2LogicalModel:parkingOccupanyDetectionType> [0..*] ?
<D2LogicalModel:parkingSecurity> D2LogicalModel:ParkingSecurityEnum
</D2LogicalModel:parkingSecurity> [0..*] ?
<D2LogicalModel:dedicatedAccess> D2LogicalModel:DedicatedAccess
</D2LogicalModel:dedicatedAccess> [0..*]
<D2LogicalModel:onlyAssignedParking>
D2LogicalModel:ParkingAssignment
</D2LogicalModel:onlyAssignedParking> [0..1] ?
<D2LogicalModel:assignedParkingAmongOthers>
D2LogicalModel:ParkingAssignment
</D2LogicalModel:assignedParkingAmongOthers> [0..1] ?
<D2LogicalModel:prohibitedParking> D2LogicalModel:ParkingAssignment
</D2LogicalModel:prohibitedParking> [0..1] ?
<D2LogicalModel:parkingEquipmentOrServiceFacility>
D2LogicalModel: ParkingSpaceBasicsEquipmentOrServiceFacilityIndexParkingEqu:
</D2LogicalModel:parkingEquipmentOrServiceFacility> [0..*] ?
<D2LogicalModel:parkingUsageScenario>
D2LogicalModel: ParkingSpaceBasicsScenarioIndexParkingUsageScenario
</D2LogicalModel:parkingUsageScenario> [0..*]
<D2LogicalModel:parkingSpaceBasicsExtension>
D2LogicalModel: ExtensionType
</D2LogicalModel:parkingSpaceBasicsExtension> [0..1]
<D2LogicalModel:identicalToParkingSpace>
D2LogicalModel:IndexReference
</D2LogicalModel:identicalToParkingSpace> [0..*] ?
<D2LogicalModel:location> D2LogicalModel:Location
</D2LogicalModel:location> [0..1]
<D2LogicalModel:parkingSpaceDimension> D2LogicalModel:Dimension
</D2LogicalModel:parkingSpaceDimension> [0..1] ?
<D2LogicalModel:parkingSpaceExtension>
D2LogicalModel: ExtensionType
</D2LogicalModel:parkingSpaceExtension> [0..1]
</...>

```

Schema Component Representation

```

<xs:complexType name="ParkingSpace">
  <xs:complexContent>
    <xs:extension base="D2LogicalModel:ParkingSpaceBasics">
      <xs:sequence>
        <xs:element name="identicalToParkingSpace"
          type="D2LogicalModel:IndexReference" minOccurs="0"
          maxOccurs="unbounded"/>
        <xs:element name="location" type="D2LogicalModel:Location"
          minOccurs="0"/>
        <xs:element name="parkingSpaceDimension"
          type="D2LogicalModel:Dimension" minOccurs="0"/>
        <xs:element name="parkingSpaceExtension"
          type="D2LogicalModel: ExtensionType" minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>

```

Complex Type: **ParkingSpaceBasics**

Super-types: None

Sub-types:

- [GroupOfParkingSpaces](#) (by extension)
- [ParkingSpace](#) (by extension)

Name	ParkingSpaceBasics
Abstract	yes
Documentation	Common properties of parking spaces and groups of parking spaces.

XML Instance Representation

```
<...>
  <D2LogicalModel:parkingSpaceOrGroupIdentifier>
    D2LogicalModel:MultilingualString
  </D2LogicalModel:parkingSpaceOrGroupIdentifier> [0..1] ?
  <D2LogicalModel:parkingFloorOrLevel> D2LogicalModel:Integer
  </D2LogicalModel:parkingFloorOrLevel> [0..1] ?
  <D2LogicalModel:accessibility> D2LogicalModel:AccessibilityEnum
  </D2LogicalModel:accessibility> [0..*] ?
  <D2LogicalModel:parkingSpaceAccessibility>
    D2LogicalModel:ParkingSpaceAccessibilityEnum
  </D2LogicalModel:parkingSpaceAccessibility> [0..*] ?
  <D2LogicalModel:parkingSpacePhysics>
    D2LogicalModel:ParkingSpacePhysicsEnum
  </D2LogicalModel:parkingSpacePhysics> [0..2] ?
  <D2LogicalModel:parkingMode> D2LogicalModel:ParkingModeEnum
  </D2LogicalModel:parkingMode> [0..1] ?
  <D2LogicalModel:parkingReservation>
    D2LogicalModel:ReservationTypeEnum
  </D2LogicalModel:parkingReservation> [0..1] ?
  <D2LogicalModel:maximumParkingDuration> D2LogicalModel:Seconds
  </D2LogicalModel:maximumParkingDuration> [0..1] ?
  <D2LogicalModel:distanceFromPrimaryRoad>
    D2LogicalModel:MetresAsNonNegativeInteger
  </D2LogicalModel:distanceFromPrimaryRoad> [0..1] ?
  <D2LogicalModel:parkingOccupanyDetectionType>
    D2LogicalModel:OccupancyDetectionTypeEnum
  </D2LogicalModel:parkingOccupanyDetectionType> [0..*] ?
  <D2LogicalModel:parkingSecurity> D2LogicalModel:ParkingSecurityEnum
  </D2LogicalModel:parkingSecurity> [0..*] ?
  <D2LogicalModel:dedicatedAccess> D2LogicalModel:DedicatedAccess
  </D2LogicalModel:dedicatedAccess> [0..*]
  <D2LogicalModel:onlyAssignedParking>
    D2LogicalModel:ParkingAssignment
  </D2LogicalModel:onlyAssignedParking> [0..1] ?
  <D2LogicalModel:assignedParkingAmongOthers>
    D2LogicalModel:ParkingAssignment
  </D2LogicalModel:assignedParkingAmongOthers> [0..1] ?
  <D2LogicalModel:prohibitedParking> D2LogicalModel:ParkingAssignment
  </D2LogicalModel:prohibitedParking> [0..1] ?
  <D2LogicalModel:parkingEquipmentOrServiceFacility>
    D2LogicalModel: ParkingSpaceBasicsEquipmentOrServiceFacilityIndexParkingEqu:
```

```

</D2LogicalModel:parkingEquipmentOrServiceFacility> [0..*] ?
<D2LogicalModel:parkingUsageScenario>
D2LogicalModel: ParkingSpaceBasicsScenarioIndexParkingUsageScenario
</D2LogicalModel:parkingUsageScenario> [0..*]
<D2LogicalModel:parkingSpaceBasicsExtension>
D2LogicalModel: ExtensionType
</D2LogicalModel:parkingSpaceBasicsExtension> [0..1]
</...>

```

Schema Component Representation

```

<xs:complexType name="ParkingSpaceBasics" abstract="true">
  <xs:sequence>
    <xs:element name="parkingSpaceOrGroupIdentifier"
      type="D2LogicalModel:MultilingualString" minOccurs="0"
      maxOccurs="1"/>
    <xs:element name="parkingFloorOrLevel"
      type="D2LogicalModel:Integer" minOccurs="0" maxOccurs="1"/>
    <xs:element name="accessibility"
      type="D2LogicalModel:AccessibilityEnum" minOccurs="0"
      maxOccurs="unbounded"/>
    <xs:element name="parkingSpaceAccessibility"
      type="D2LogicalModel:ParkingSpaceAccessibilityEnum"
      minOccurs="0" maxOccurs="unbounded"/>
    <xs:element name="parkingSpacePhysics"
      type="D2LogicalModel:ParkingSpacePhysicsEnum" minOccurs="0"
      maxOccurs="2"/>
    <xs:element name="parkingMode"
      type="D2LogicalModel:ParkingModeEnum" minOccurs="0"
      maxOccurs="1"/>
    <xs:element name="parkingReservation"
      type="D2LogicalModel:ReservationTypeEnum" minOccurs="0"
      maxOccurs="1"/>
    <xs:element name="maximumParkingDuration"
      type="D2LogicalModel:Seconds" minOccurs="0" maxOccurs="1"/>
    <xs:element name="distanceFromPrimaryRoad"
      type="D2LogicalModel:MetresAsNonNegativeInteger" minOccurs="0"
      maxOccurs="1"/>
    <xs:element name="parkingOccupanyDetectionType"
      type="D2LogicalModel:OccupancyDetectionTypeEnum" minOccurs="0"
      maxOccurs="unbounded"/>
    <xs:element name="parkingSecurity"
      type="D2LogicalModel:ParkingSecurityEnum" minOccurs="0"
      maxOccurs="unbounded"/>
    <xs:element name="dedicatedAccess"
      type="D2LogicalModel:DedicatedAccess" minOccurs="0"
      maxOccurs="unbounded"/>
    <xs:element name="onlyAssignedParking"
      type="D2LogicalModel:ParkingAssignment" minOccurs="0"/>
    <xs:element name="assignedParkingAmongOthers"
      type="D2LogicalModel:ParkingAssignment" minOccurs="0"/>
    <xs:element name="prohibitedParking"
      type="D2LogicalModel:ParkingAssignment" minOccurs="0"/>
    <xs:element name="parkingEquipmentOrServiceFacility"
      type="D2LogicalModel: ParkingSpaceBasicsEquipmentOrServiceFacilityIndexPa
      minOccurs="0" maxOccurs="unbounded"/>
    <xs:element name="parkingUsageScenario"
      type="D2LogicalModel: ParkingSpaceBasicsScenarioIndexParkingUsageScenario
      minOccurs="0" maxOccurs="unbounded"/>
    <xs:element name="parkingSpaceBasicsExtension"
      type="D2LogicalModel: ExtensionType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>

```

Complex Type: **ParkingStandardsAndSecurity**

Super-types: None

Sub-types: None

Name ParkingStandardsAndSecurity

Abstract no

Documentation Security measures and standards or standard-like categorization for a parking site.

XML Instance Representation

```
<...>
  <D2LogicalModel:labelSecurityLevel>
    D2LogicalModel:LABELSecurityLevelEnum
  </D2LogicalModel:labelSecurityLevel> [0..1] ?
  <D2LogicalModel:labelServiceLevel>
    D2LogicalModel:LABELServiceLevelEnum
  </D2LogicalModel:labelServiceLevel> [0..1] ?
  <D2LogicalModel:labelSecurityLevelSelfAssessment>
    D2LogicalModel:LABELSecurityLevelEnum
  </D2LogicalModel:labelSecurityLevelSelfAssessment> [0..1] ?
  <D2LogicalModel:labelServiceLevelSelfAssessment>
    D2LogicalModel:LABELServiceLevelEnum
  </D2LogicalModel:labelServiceLevelSelfAssessment> [0..1] ?
  <D2LogicalModel:parkingSecurity> D2LogicalModel:ParkingSecurityEnum
</D2LogicalModel:parkingSecurity> [1..*] ?
  <D2LogicalModel:parkingAdditionalSecurity>
    D2LogicalModel:MultilingualString
  </D2LogicalModel:parkingAdditionalSecurity> [0..*] ?
  <D2LogicalModel:parkingSupervision>
    D2LogicalModel:ParkingSupervisionEnum
  </D2LogicalModel:parkingSupervision> [0..*] ?
  <D2LogicalModel:parkingSecurityNationalClassification>
    D2LogicalModel:MultilingualString
  </D2LogicalModel:parkingSecurityNationalClassification> [0..1] ?
  <D2LogicalModel:certifiedSecureParking> D2LogicalModel:Boolean
</D2LogicalModel:certifiedSecureParking> [0..1] ?
  <D2LogicalModel:dateOfCertification> D2LogicalModel:Date
</D2LogicalModel:dateOfCertification> [0..1] ?
  <D2LogicalModel:parkingStandardsAndSecurityExtension>
    D2LogicalModel:ExtensionType
  </D2LogicalModel:parkingStandardsAndSecurityExtension> [0..1]
</...>
```

Schema Component Representation

```
<xs:complexType name="ParkingStandardsAndSecurity">
  <xs:sequence>
    <xs:element name="labelSecurityLevel"
      type="D2LogicalModel:LABELSecurityLevelEnum" minOccurs="0"
      maxOccurs="1"/>
    <xs:element name="labelServiceLevel"
      type="D2LogicalModel:LABELServiceLevelEnum" minOccurs="0"
      maxOccurs="1"/>
    <xs:element name="labelSecurityLevelSelfAssessment"
      type="D2LogicalModel:LABELSecurityLevelEnum" minOccurs="0"
      maxOccurs="1"/>
```

```

maxOccurs="1"/>
<xs:element name="labelServiceLevelSelfAssessment"
type="D2LogicalModel:LABELServiceLevelEnum" minOccurs="0"
maxOccurs="1"/>
<xs:element name="parkingSecurity"
type="D2LogicalModel:ParkingSecurityEnum" minOccurs="1"
maxOccurs="unbounded"/>
<xs:element name="parkingAdditionalSecurity"
type="D2LogicalModel:MultilingualString" minOccurs="0"
maxOccurs="unbounded"/>
<xs:element name="parkingSupervision"
type="D2LogicalModel:ParkingSupervisionEnum" minOccurs="0"
maxOccurs="unbounded"/>
<xs:element name="parkingSecurityNationalClassification"
type="D2LogicalModel:MultilingualString" minOccurs="0"
maxOccurs="1"/>
<xs:element name="certifiedSecureParking"
type="D2LogicalModel:Boolean" minOccurs="0" maxOccurs="1"/>
<xs:element name="dateOfCertification"
type="D2LogicalModel:Date" minOccurs="0" maxOccurs="1"/>
<xs:element name="parkingStandardsAndSecurityExtension"
type="D2LogicalModel:ExtensionType" minOccurs="0"/>
</xs:sequence>
</xs:complexType>

```

[top](#)

Complex Type: **ParkingStatusColourMapping**

Super-types: None

Sub-types: None

Name	ParkingStatusColourMapping
Abstract	no
Documentation	Defines a pair of 'parkingSiteStatus' and a corresponding colour.

XML Instance Representation

```

<...>
  <D2LogicalModel:parkingSiteStatus>
    D2LogicalModel:ParkingSiteStatusEnum
  </D2LogicalModel:parkingSiteStatus> [1] ?
  <D2LogicalModel:rgbColour> D2LogicalModel:RGBColour
  </D2LogicalModel:rgbColour> [1]
  <D2LogicalModel:parkingStatusColourMappingExtension>
    D2LogicalModel:ExtensionType
  </D2LogicalModel:parkingStatusColourMappingExtension> [0..1]
</...>

```

Schema Component Representation

```

<xs:complexType name="ParkingStatusColourMapping">
  <xs:sequence>
    <xs:element name="parkingSiteStatus"
type="D2LogicalModel:ParkingSiteStatusEnum" minOccurs="1"
maxOccurs="1"/>
    <xs:element name="rgbColour" type="D2LogicalModel:RGBColour"/>
    <xs:element name="parkingStatusColourMappingExtension"
type="D2LogicalModel:ExtensionType" minOccurs="0"/>
  
```

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

[top](#)

Complex Type: **ParkingTable**

Super-types: None

Sub-types: None

Name ParkingTable

Abstract no

Documentation A collection of parking records, which can be parking sites or groups of parking sites.

XML Instance Representation

```
<...
  id="xs:string [1]"
  version="xs:string [1]">
    <D2LogicalModel:parkingTableName> D2LogicalModel:MultilingualString
  </D2LogicalModel:parkingTableName> [0..1] ?
    <D2LogicalModel:parkingTableVersionTime> D2LogicalModel:DateTime
  </D2LogicalModel:parkingTableVersionTime> [1] ?
    <D2LogicalModel:parkingRecord> D2LogicalModel:ParkingRecord
  </D2LogicalModel:parkingRecord> [1..*]
    <D2LogicalModel:parkingTableExtension>
      D2LogicalModel: _ExtensionType
    </D2LogicalModel:parkingTableExtension> [0..1]
  </...>
```

Schema Component Representation

```
<xs:complexType name="ParkingTable">
  <xs:sequence>
    <xs:element name="parkingTableName"
      type="D2LogicalModel:MultilingualString" minOccurs="0"
      maxOccurs="1"/>
    <xs:element name="parkingTableVersionTime"
      type="D2LogicalModel:DateTime" minOccurs="1" maxOccurs="1"/>
    <xs:element name="parkingRecord"
      type="D2LogicalModel:ParkingRecord" maxOccurs="unbounded"/>
    <xs:element name="parkingTableExtension"
      type="D2LogicalModel: _ExtensionType" minOccurs="0"/>
  </xs:sequence>
  <xs:attribute name="id" type="xs:string" use="required"/>
  <xs:attribute name="version" type="xs:string" use="required"/>
</xs:complexType>
```

[top](#)

Complex Type: **ParkingTablePublication**

Super-types: None

Sub-types: None

Name	ParkingTablePublication
Abstract	no
Documentation	A publication defining one or more tables that have entries of parking sites or groups of them, located in an urban or interurban context.

XML Instance Representation

```
<...>
  <D2LogicalModel:headerInformation> D2LogicalModel:HeaderInformation
</D2LogicalModel:headerInformation> [0..1]
  <D2LogicalModel:parkingTable> D2LogicalModel:ParkingTable
</D2LogicalModel:parkingTable> [1..*]
</...>
```

Schema Component Representation

```
<xs:complexType name="ParkingTablePublication">
  <xs:sequence>
    <xs:element name="headerInformation"
      type="D2LogicalModel:HeaderInformation" minOccurs="0"/>
    <xs:element name="parkingTable"
      type="D2LogicalModel:ParkingTable" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>
```

[top](#)

Complex Type: **ParkingThresholds**

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

Name	ParkingThresholds
Abstract	no
Documentation	Configuration parameters of the parking site, used among others for the dynamic attribute 'parkingStatus'. This component or all elements of it can be overridden in the dynamic model.

XML Instance Representation

```
<...>
  <D2LogicalModel:almostFullDecreasing>
    D2LogicalModel:NonNegativeInteger
  </D2LogicalModel:almostFullDecreasing> [0..1] ?
  <D2LogicalModel:almostFullIncreasing>
    D2LogicalModel:NonNegativeInteger
  </D2LogicalModel:almostFullIncreasing> [0..1] ?
  <D2LogicalModel:entranceFull> D2LogicalModel:NonNegativeInteger
</D2LogicalModel:entranceFull> [0..1] ?
  <D2LogicalModel:fullDecreasing> D2LogicalModel:NonNegativeInteger
</D2LogicalModel:fullDecreasing> [0..1] ?
  <D2LogicalModel:fullIncreasing> D2LogicalModel:NonNegativeInteger
</D2LogicalModel:fullIncreasing> [0..1] ?
  <D2LogicalModel:overcrowding> D2LogicalModel:NonNegativeInteger
</D2LogicalModel:overcrowding> [0..1] ?
  <D2LogicalModel:overcrowdingLevel1>
```

```

D2LogicalModel:NonNegativeInteger
</D2LogicalModel:overcrowdingLevel1> [0..1] ?
<D2LogicalModel:overcrowdingLevel2>
D2LogicalModel:NonNegativeInteger
</D2LogicalModel:overcrowdingLevel2> [0..1] ?
<D2LogicalModel:parkingLastMaximumOccupancy>
D2LogicalModel:NonNegativeInteger
</D2LogicalModel:parkingLastMaximumOccupancy> [0..1] ?
<D2LogicalModel:parkingStatusColourMapping>
D2LogicalModel:ParkingStatusColourMapping
</D2LogicalModel:parkingStatusColourMapping> [0..*]
<D2LogicalModel:parkingThresholdsExtension>
D2LogicalModel:_ExtensionType
</D2LogicalModel:parkingThresholdsExtension> [0..1]
</...>

```

Schema Component Representation

```

<xs:complexType name="ParkingThresholds">
  <xs:sequence>
    <xs:element name="almostFullDecreasing"
      type="D2LogicalModel:NonNegativeInteger" minOccurs="0"
      maxOccurs="1"/>
    <xs:element name="almostFullIncreasing"
      type="D2LogicalModel:NonNegativeInteger" minOccurs="0"
      maxOccurs="1"/>
    <xs:element name="entranceFull"
      type="D2LogicalModel:NonNegativeInteger" minOccurs="0"
      maxOccurs="1"/>
    <xs:element name="fullDecreasing"
      type="D2LogicalModel:NonNegativeInteger" minOccurs="0"
      maxOccurs="1"/>
    <xs:element name="fullIncreasing"
      type="D2LogicalModel:NonNegativeInteger" minOccurs="0"
      maxOccurs="1"/>
    <xs:element name="overcrowding"
      type="D2LogicalModel:NonNegativeInteger" minOccurs="0"
      maxOccurs="1"/>
    <xs:element name="overcrowdingLevel1"
      type="D2LogicalModel:NonNegativeInteger" minOccurs="0"
      maxOccurs="1"/>
    <xs:element name="overcrowdingLevel2"
      type="D2LogicalModel:NonNegativeInteger" minOccurs="0"
      maxOccurs="1"/>
    <xs:element name="parkingLastMaximumOccupancy"
      type="D2LogicalModel:NonNegativeInteger" minOccurs="0"
      maxOccurs="1"/>
    <xs:element name="parkingStatusColourMapping"
      type="D2LogicalModel:ParkingStatusColourMapping" minOccurs="0"
      maxOccurs="unbounded"/>
    <xs:element name="parkingThresholdsExtension"
      type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>

```

[top](#)

Complex Type: **ParkingUsageScenario**

Super-types: None

Sub-types: None

Name ParkingUsageScenario

Abstract no

Documentation A special type of usage available for the parking site or the group of parking spaces. In the 'ParkingStatusPublication', the operation type (in operation or not) can be defined.

XML Instance Representation

```
<...>
  <D2LogicalModel:parkingUsageScenario>
    D2LogicalModel:ParkingUsageScenarioEnum
  </D2LogicalModel:parkingUsageScenario> [1] ?
  <D2LogicalModel:truckParkingDynamicManagement>
    D2LogicalModel:TruckParkingDynamicManagementEnum
  </D2LogicalModel:truckParkingDynamicManagement> [0..*] ?
  <D2LogicalModel:scenarioAvailability> D2LogicalModel:OverallPeriod
  </D2LogicalModel:scenarioAvailability> [0..1]
  <D2LogicalModel:parkingUsageScenarioExtension>
    D2LogicalModel:_ExtensionType
  </D2LogicalModel:parkingUsageScenarioExtension> [0..1]
</...>
```

Schema Component Representation

```
<xs:complexType name="ParkingUsageScenario">
  <xs:sequence>
    <xs:element name="parkingUsageScenario"
      type="D2LogicalModel:ParkingUsageScenarioEnum" minOccurs="1"
      maxOccurs="1"/>
    <xs:element name="truckParkingDynamicManagement"
      type="D2LogicalModel:TruckParkingDynamicManagementEnum"
      minOccurs="0" maxOccurs="unbounded"/>
    <xs:element name="scenarioAvailability"
      type="D2LogicalModel:OverallPeriod" minOccurs="0"/>
    <xs:element name="parkingUsageScenarioExtension"
      type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
```

[top](#)

Complex Type: ParkingVMS

Super-types: None

Sub-types: None

Name ParkingVMS

Abstract no

Documentation A reference to a record that contains the metadata for a specific VMS unit that may be used to manage the parking site (e.g. to indicate to drivers the current availability of spaces).

XML Instance Representation

```
<...>
```

```

<D2LogicalModel:vmsUnitUsedToManageParking>
D2LogicalModel: VmsUnitRecordVersionedReference
</D2LogicalModel:vmsUnitUsedToManageParking> [1] ?
<D2LogicalModel:vmsOperator> D2LogicalModel:Contact
</D2LogicalModel:vmsOperator> [0..*]
<D2LogicalModel:parkingVMSExtension> D2LogicalModel: ExtensionType
</D2LogicalModel:parkingVMSExtension> [0..1]
</...>

```

Schema Component Representation

```

<xs:complexType name="ParkingVMS">
  <xs:sequence>
    <xs:element name="vmsUnitUsedToManageParking"
      type="D2LogicalModel: VmsUnitRecordVersionedReference"
      minOccurs="1" maxOccurs="1"/>
    <xs:element name="vmsOperator" type="D2LogicalModel:Contact"
      minOccurs="0" maxOccurs="unbounded"/>
    <xs:element name="parkingVMSExtension"
      type="D2LogicalModel: ExtensionType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>

```

[top](#)

Complex Type: **PayloadPublication**

Super-types: None

Sub-types:

- [GenericPublication](#) (by extension)

Name	PayloadPublication
Abstract	yes
Documentation	A payload publication of traffic related information or associated management information created at a specific point in time that can be exchanged via a DATEX II interface.

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: **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: **Period**

Super-types: None

Sub-types: None

Name Period

Abstract no

Documentation A continuous time period or a set of discontinuous time periods defined by the intersection of a set of criteria all within an overall delimiting interval.

XML Instance Representation

```
<...>
  <D2LogicalModel:startOfPeriod> D2LogicalModel:DateTime
</D2LogicalModel:startOfPeriod> [0..1] ?
  <D2LogicalModel:endOfPeriod> D2LogicalModel:DateTime
</D2LogicalModel:endOfPeriod> [0..1] ?
  <D2LogicalModel:periodName> D2LogicalModel:MultilingualString
</D2LogicalModel:periodName> [0..1] ?
  <D2LogicalModel:recurringTimePeriodOfDay>
D2LogicalModel:TimePeriodOfDay
</D2LogicalModel:recurringTimePeriodOfDay> [0..*] ?
  <D2LogicalModel:recurringDayWeekMonthPeriod>
D2LogicalModel:DayWeekMonth
</D2LogicalModel:recurringDayWeekMonthPeriod> [0..*] ?
  <D2LogicalModel:periodExtension>
D2LogicalModel:_PeriodExtensionType
</D2LogicalModel:periodExtension> [0..1]
</...>
```

Schema Component Representation

```
<xs:complexType name="Period">
  <xs:sequence>
    <xs:element name="startOfPeriod" type="D2LogicalModel:DateTime"
minOccurs="0" maxOccurs="1"/>
    <xs:element name="endOfPeriod" type="D2LogicalModel:DateTime"
minOccurs="0" maxOccurs="1"/>
    <xs:element name="periodName"
type="D2LogicalModel:MultilingualString" minOccurs="0"
maxOccurs="1"/>
    <xs:element name="recurringTimePeriodOfDay"
type="D2LogicalModel:TimePeriodOfDay" minOccurs="0"
maxOccurs="unbounded"/>
    <xs:element name="recurringDayWeekMonthPeriod"
type="D2LogicalModel:DayWeekMonth" minOccurs="0"
maxOccurs="unbounded"/>
    <xs:element name="periodExtension"
type="D2LogicalModel:_PeriodExtensionType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
```

[top](#)

Complex Type: **PeriodExtended**

Super-types: None

Sub-types: None

Name PeriodExtended

Abstract	no
Documentation	An extension point for Period offering the possibility to describe special days and public holidays.

XML Instance Representation

```
<...>
  <D2LogicalModel:recurringSpecialDay> D2LogicalModel:SpecialDay
</D2LogicalModel:recurringSpecialDay> [0..*] ?
</...>
```

Schema Component Representation

```
<xs:complexType name="PeriodExtended">
  <xs:sequence>
    <xs:element name="recurringSpecialDay"
      type="D2LogicalModel:SpecialDay" minOccurs="0"
      maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>
```

[top](#)

Complex Type: **PermitsAndProhibitions**

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

Name	PermitsAndProhibitions
Abstract	no
Documentation	Defines sets of action and regulations to specify permitted and prohibited issues.

XML Instance Representation

```
<...>
  <D2LogicalModel:activity> D2LogicalModel:RestAreaActivityEnum
</D2LogicalModel:activity> [1] ?
  <D2LogicalModel:regulation> D2LogicalModel:RegulationEnum
</D2LogicalModel:regulation> [1] ?
  <D2LogicalModel:permitsAndProhibitionsExtension>
    D2LogicalModel:ExtensionType
  </D2LogicalModel:permitsAndProhibitionsExtension> [0..1]
</...>
```

Schema Component Representation

```

<xs:complexType name="PermitsAndProhibitions">
  <xs:sequence>
    <xs:element name="activity"
      type="D2LogicalModel:RestAreaActivityEnum" minOccurs="1"
      maxOccurs="1"/>
    <xs:element name="regulation"
      type="D2LogicalModel:RegulationEnum" minOccurs="1"
      maxOccurs="1"/>
    <xs:element name="permitsAndProhibitionsExtension"
      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
<u>Abstract</u>	no
Documentation	A single geospatial point.

XML Instance Representation

```

<...>
  <D2LogicalModel:groupOfLocationsExtension>
    D2LogicalModel:_ExtensionType
  </D2LogicalModel:groupOfLocationsExtension> [0..1]
  <D2LogicalModel:externalReferencing>
    D2LogicalModel:ExternalReferencing
  </D2LogicalModel:externalReferencing> [0..*]
  <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:destination> D2LogicalModel:Destination
  </D2LogicalModel:destination> [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>
```

[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: **PointDestination**

Super-types: [Destination](#) < **PointDestination** (by extension)

Sub-types: None

Name PointDestination

Abstract no

Documentation The specification of the destination of a defined route or itinerary which is a point.

XML Instance Representation

```
<...>
```

```

<D2LogicalModel:destinationExtension> D2LogicalModel: _ExtensionType
</D2LogicalModel:destinationExtension> [0..1]
<D2LogicalModel:point> D2LogicalModel:Point </D2LogicalModel:point>
[1]
<D2LogicalModel:pointDestinationExtension>
D2LogicalModel: _ExtensionType
</D2LogicalModel:pointDestinationExtension> [0..1]
</...>

```

Schema Component Representation

```

<xs:complexType name="PointDestination">
  <xs:complexContent>
    <xs:extension base="D2LogicalModel:Destination">
      <xs:sequence>
        <xs:element name="point" type="D2LogicalModel:Point"/>
        <xs:element name="pointDestinationExtension"
          type="D2LogicalModel: _ExtensionType" minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

```

[top](#)

Complex Type: **PointExtended**

Super-types: None

Sub-types: None

Name	PointExtended
<u>Abstract</u>	no
Documentation	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: **PolygonArea**

Super-types: None

Sub-types: None

Name PolygonArea

Abstract no

Documentation defines points for a closed polygon-shape describing the area

XML Instance Representation

```
<...>
  <D2LogicalModel:sectionName> D2LogicalModel:MultilingualString
</D2LogicalModel:sectionName> [0..1] ?
  <D2LogicalModel:pointCoordinates>
    D2LogicalModel:PolygonAreaIndexPointCoordinates
  </D2LogicalModel:pointCoordinates> [0..*]
  <D2LogicalModel:polygonAreaExtension> D2LogicalModel:_ExtensionType
</D2LogicalModel:polygonAreaExtension> [0..1]
</...>
```

Schema Component Representation

```
<xs:complexType name="PolygonArea">
  <xs:sequence>
    <xs:element name="sectionName"
      type="D2LogicalModel:MultilingualString" minOccurs="0"
      maxOccurs="1"/>
    <xs:element name="pointCoordinates"
      type="D2LogicalModel:_PolygonAreaIndexPointCoordinates"
      minOccurs="0" maxOccurs="unbounded"/>
    <xs:element name="polygonAreaExtension"
      type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
```

[top](#)

Complex Type: **PublicHoliday**

Super-types: None

Sub-types: None

Name PublicHoliday

Abstract no

Documentation Specification of the public holiday type in a specific country or region. Use this component only when specialDayType is set to 'publicHoliday' or 'holidays'.

XML Instance Representation

```
<...>
  <D2LogicalModel:country> D2LogicalModel:CountryEnum
</D2LogicalModel:country> [1] ?
  <D2LogicalModel:countrySubdivision> D2LogicalModel:String
</D2LogicalModel:countrySubdivision> [0..1] ?
</...>
```

```

<D2LogicalModel:region> D2LogicalModel:MultilingualString
</D2LogicalModel:region> [0..1] ?
<D2LogicalModel:publicHolidayType>
D2LogicalModel:PublicHolidayTypeEnum
</D2LogicalModel:publicHolidayType> [1] ?
<D2LogicalModel:publicHolidayName>
D2LogicalModel:MultilingualString
</D2LogicalModel:publicHolidayName> [0..1] ?
<D2LogicalModel:publicHolidayExtension>
D2LogicalModel:_ExtensionType
</D2LogicalModel:publicHolidayExtension> [0..1]
</...>

```

Schema Component Representation

```

<xs:complexType name="PublicHoliday">
  <xs:sequence>
    <xs:element name="country" type="D2LogicalModel:CountryEnum"
      minOccurs="1" maxOccurs="1"/>
    <xs:element name="countrySubdivision"
      type="D2LogicalModel:String" minOccurs="0" maxOccurs="1"/>
    <xs:element name="region"
      type="D2LogicalModel:MultilingualString" minOccurs="0"
      maxOccurs="1"/>
    <xs:element name="publicHolidayType"
      type="D2LogicalModel:PublicHolidayTypeEnum" minOccurs="1"
      maxOccurs="1"/>
    <xs:element name="publicHolidayName"
      type="D2LogicalModel:MultilingualString" minOccurs="0"
      maxOccurs="1"/>
    <xs:element name="publicHolidayExtension"
      type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>

```

[top](#)

Complex Type: **RGBColour**

Super-types: None

Sub-types: None

Name RGBColour

Abstract no

Documentation An RGB colour described by values for red, green and blue (0..255) as well as an optional name.

XML Instance Representation

```

<...>
  <D2LogicalModel:rgbRedValue> D2LogicalModel:NonNegativeInteger
  </D2LogicalModel:rgbRedValue> [1] ?
  <D2LogicalModel:rgbGreenValue> D2LogicalModel:NonNegativeInteger
  </D2LogicalModel:rgbGreenValue> [1] ?
  <D2LogicalModel:rgbBlueValue> D2LogicalModel:NonNegativeInteger
  </D2LogicalModel:rgbBlueValue> [1] ?
  <D2LogicalModel:colourName> D2LogicalModel:MultilingualString
  </D2LogicalModel:colourName> [0..1] ?
  <D2LogicalModel:rgbColourExtension> D2LogicalModel:_ExtensionType

```

```
</D2LogicalModel:rgbColourExtension> [0..1]
</...>
```

Schema Component Representation

```
<xs:complexType name="RGBColour">
  <xs:sequence>
    <xs:element name="rgbRedValue"
      type="D2LogicalModel:NonNegativeInteger" minOccurs="1"
      maxOccurs="1"/>
    <xs:element name="rgbGreenValue"
      type="D2LogicalModel:NonNegativeInteger" minOccurs="1"
      maxOccurs="1"/>
    <xs:element name="rgbBlueValue"
      type="D2LogicalModel:NonNegativeInteger" minOccurs="1"
      maxOccurs="1"/>
    <xs:element name="colourName"
      type="D2LogicalModel:MultilingualString" minOccurs="0"
      maxOccurs="1"/>
    <xs:element name="rgbColourExtension"
      type="D2LogicalModel:ExtensionType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
```

[top](#)

Complex Type: Reference

Super-types: None

Sub-types:

- [ParkingAccessReference](#) (by extension)

Name Reference

Abstract no

XML Instance Representation

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

Schema Component Representation

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

[top](#)

Complex Type: Referent

Super-types: None

Sub-types: None

Name Referent

Abstract no

Documentation

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:

- [RoadNode](#) (by extension)

Name Road

Abstract no

Documentation Identification of a road by its name, identifier, type ...

XML Instance Representation

```
<...>
  <D2LogicalModel:nameOfRoad> D2LogicalModel:MultilingualString
```

```

</D2LogicalModel:nameOfRoad> [0..1] ?
<D2LogicalModel:roadIdentifier> D2LogicalModel:MultilingualString
</D2LogicalModel:roadIdentifier> [1] ?
<D2LogicalModel:typeOfRoad> D2LogicalModel:RoadTypeEnum
</D2LogicalModel:typeOfRoad> [0..1] ?
<D2LogicalModel:roadDestination> D2LogicalModel:MultilingualString
</D2LogicalModel:roadDestination> [1..*] ?
<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="1"
      maxOccurs="1"/>
    <xs:element name="typeOfRoad" type="D2LogicalModel:RoadTypeEnum"
      minOccurs="0" maxOccurs="1"/>
    <xs:element name="roadDestination"
      type="D2LogicalModel:MultilingualString" minOccurs="1"
      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> [1] ?
<D2LogicalModel:typeOfRoad> D2LogicalModel:RoadTypeEnum
</D2LogicalModel:typeOfRoad> [0..1] ?
<D2LogicalModel:roadDestination> D2LogicalModel:MultilingualString
</D2LogicalModel:roadDestination> [1..*] ?
<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: ServiceFacility

Super-types: [ParkingEquipmentOrServiceFacility](#) < **ServiceFacility** (by extension)
 Sub-types: None

Name	ServiceFacility
<u>Abstract</u>	no
Documentation	One type of service facility that is available on the parking site or located next to it. You can specify the number of this service facility type (e.g. 5 restaurants) as well as the number of subitems (e.g. 200 restaurant places).

XML Instance Representation

```

<...>
  <D2LogicalModel:equipmentOrServiceFacilityIdentifier>
    D2LogicalModel:String
  </D2LogicalModel:equipmentOrServiceFacilityIdentifier> [0..*] ?
  <D2LogicalModel:availability> D2LogicalModel:AvailabilityEnum
  </D2LogicalModel:availability> [0..1] ?
  <D2LogicalModel:numberOfEquipmentOrServiceFacility>
    D2LogicalModel:NonNegativeInteger

```

```

</D2LogicalModel:numberOfEquipmentOrServiceFacility> [0..1] ?
<D2LogicalModel:additionalDescription>
D2LogicalModel:MultilingualString
</D2LogicalModel:additionalDescription> [0..1] ?
<D2LogicalModel:otherEquipmentOrServiceFacility>
D2LogicalModel:MultilingualString
</D2LogicalModel:otherEquipmentOrServiceFacility> [0..1] ?
<D2LogicalModel:accessibility> D2LogicalModel:AccessibilityEnum
</D2LogicalModel:accessibility> [0..*] ?
<D2LogicalModel:nameOrBrand> D2LogicalModel:MultilingualString
</D2LogicalModel:nameOrBrand> [0..1] ?
<D2LogicalModel:comment> D2LogicalModel:MultilingualString
</D2LogicalModel:comment> [0..1] ?
<D2LogicalModel:photoUrl> D2LogicalModel:Url
</D2LogicalModel:photoUrl> [0..1] ?
<D2LogicalModel:applicableForUser> D2LogicalModel:UserTypeEnum
</D2LogicalModel:applicableForUser> [0..*] ?
<D2LogicalModel:availabilityAndOpeningTimes>
D2LogicalModel:OpeningTimes
</D2LogicalModel:availabilityAndOpeningTimes> [0..1] ?
<D2LogicalModel:tariffsAndPayment> D2LogicalModel:TariffsAndPayment
</D2LogicalModel:tariffsAndPayment> [0..1]
<D2LogicalModel:groupOfLocations> D2LogicalModel:GroupOfLocations
</D2LogicalModel:groupOfLocations> [0..1]
<D2LogicalModel:applicableForVehicles>
D2LogicalModel:VehicleCharacteristics
</D2LogicalModel:applicableForVehicles> [0..*]
<D2LogicalModel:parkingEquipmentOrServiceFacilityExtension>
D2LogicalModel:ExtensionType
</D2LogicalModel:parkingEquipmentOrServiceFacilityExtension> [0..1]
<D2LogicalModel:serviceFacilityType>
D2LogicalModel:ServiceFacilityTypeEnum
</D2LogicalModel:serviceFacilityType> [1] ?
<D2LogicalModel:numberOfSubitems> D2LogicalModel:NonNegativeInteger
</D2LogicalModel:numberOfSubitems> [0..1] ?
<D2LogicalModel:distanceFromParkingSite>
D2LogicalModel:MetresAsNonNegativeInteger
</D2LogicalModel:distanceFromParkingSite> [0..1] ?
<D2LogicalModel:serviceFacilityExtension>
D2LogicalModel:ExtensionType
</D2LogicalModel:serviceFacilityExtension> [0..1]
</...>

```

Schema Component Representation

```

<xs:complexType name="ServiceFacility">
  <xs:complexContent>
    <xs:extension
      base="D2LogicalModel:ParkingEquipmentOrServiceFacility">
      <xs:sequence>
        <xs:element name="serviceFacilityType"
          type="D2LogicalModel:ServiceFacilityTypeEnum" minOccurs="1"
          maxOccurs="1"/>
        <xs:element name="numberOfSubitems"
          type="D2LogicalModel:NonNegativeInteger" minOccurs="0"
          maxOccurs="1"/>
        <xs:element name="distanceFromParkingSite"
          type="D2LogicalModel:MetresAsNonNegativeInteger"
          minOccurs="0" maxOccurs="1"/>
        <xs:element name="serviceFacilityExtension"
          type="D2LogicalModel:ExtensionType" minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </complexContent>
</complexType>

```

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

[top](#)

Complex Type: **SpecialDay**

Super-types: None

Sub-types: None

Name SpecialDay

Abstract no

Documentation Specification of a special day, for example schoolDay, electionDay, ... Gives also the possibility to define a public holiday (country specific).

XML Instance Representation

```
<...>
  <D2LogicalModel:intersectWithApplicableDays> D2LogicalModel:Boolean
</D2LogicalModel:intersectWithApplicableDays> [1] ?
  <D2LogicalModel:specialDayType> D2LogicalModel:SpecialDayTypeEnum
</D2LogicalModel:specialDayType> [1] ?
  <D2LogicalModel:specialDayName> D2LogicalModel:MultilingualString
</D2LogicalModel:specialDayName> [0..1] ?
  <D2LogicalModel:publicHoliday> D2LogicalModel:PublicHoliday
</D2LogicalModel:publicHoliday> [0..*]
  <D2LogicalModel:specialDayExtension> D2LogicalModel:ExtensionType
</D2LogicalModel:specialDayExtension> [0..1]
</...>
```

Schema Component Representation

```
<xs:complexType name="SpecialDay">
  <xs:sequence>
    <xs:element name="intersectWithApplicableDays"
      type="D2LogicalModel:Boolean" minOccurs="1" maxOccurs="1"/>
    <xs:element name="specialDayType"
      type="D2LogicalModel:SpecialDayTypeEnum" minOccurs="1"
      maxOccurs="1"/>
    <xs:element name="specialDayName"
      type="D2LogicalModel:MultilingualString" minOccurs="0"
      maxOccurs="1"/>
    <xs:element name="publicHoliday"
      type="D2LogicalModel:PublicHoliday" minOccurs="0"
      maxOccurs="unbounded"/>
    <xs:element name="specialDayExtension"
      type="D2LogicalModel:ExtensionType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
```

[top](#)

Complex Type: **SpecialLocationParkingSite**

Super-types: [ParkingRecord](#) < [ParkingSite](#) (by extension) <
SpecialLocationParkingSite (by extension)

Sub-types: None

Name	SpecialLocationParkingSite
Abstract	no
Documentation	A parking site which is located at a special location, often associated with some building.

XML Instance Representation

```
<...  
  id="xs:string [1]"  
  version="xs:string [1]">  
    <D2LogicalModel:parkingName> D2LogicalModel:MultilingualString  
    </D2LogicalModel:parkingName> [1] ?  
    <D2LogicalModel:parkingAlias> D2LogicalModel:MultilingualString  
    </D2LogicalModel:parkingAlias> [0..*] ?  
    <D2LogicalModel:parkingDescription>  
    D2LogicalModel:MultilingualString  
    </D2LogicalModel:parkingDescription> [0..1] ?  
    <D2LogicalModel:parkingRecordVersionTime> D2LogicalModel:DateTime  
    </D2LogicalModel:parkingRecordVersionTime> [1] ?  
    <D2LogicalModel:parkingNumberOfSpaces>  
    D2LogicalModel:NonNegativeInteger  
    </D2LogicalModel:parkingNumberOfSpaces> [0..1] ?  
    <D2LogicalModel:parkingPrincipalNumberOfSpaces>  
    D2LogicalModel:NonNegativeInteger  
    </D2LogicalModel:parkingPrincipalNumberOfSpaces> [0..1] ?  
    <D2LogicalModel:maximumParkingDuration> D2LogicalModel:Seconds  
    </D2LogicalModel:maximumParkingDuration> [0..1] ?  
    <D2LogicalModel:photoUrl> D2LogicalModel:Url  
    </D2LogicalModel:photoUrl> [0..1] ?  
    <D2LogicalModel:urlLinkAddress> D2LogicalModel:Url  
    </D2LogicalModel:urlLinkAddress> [0..1] ?  
    <D2LogicalModel:parkingOccupancyDetectionType>  
    D2LogicalModel:OccupancyDetectionTypeEnum  
    </D2LogicalModel:parkingOccupancyDetectionType> [0..*] ?  
    <D2LogicalModel:emergencyContact> D2LogicalModel:Contact  
    </D2LogicalModel:emergencyContact> [0..*] ?  
    <D2LogicalModel:owner> D2LogicalModel:Contact  
    </D2LogicalModel:owner> [0..*] ?  
    <D2LogicalModel:responisbleAuthority> D2LogicalModel:Contact  
    </D2LogicalModel:responisbleAuthority> [0..*] ?  
    <D2LogicalModel:securityService> D2LogicalModel:Contact  
    </D2LogicalModel:securityService> [0..*] ?  
    <D2LogicalModel:operator> D2LogicalModel:Contact  
    </D2LogicalModel:operator> [1..*] ?  
    <D2LogicalModel:servicePartner> D2LogicalModel:Contact  
    </D2LogicalModel:servicePartner> [0..*] ?  
    <D2LogicalModel:parkingVMS> D2LogicalModel:ParkingVMS  
    </D2LogicalModel:parkingVMS> [0..*]  
    <D2LogicalModel:parkingLocation> D2LogicalModel:GroupOfLocations  
    </D2LogicalModel:parkingLocation> [1] ?  
    <D2LogicalModel:parkingRoute> D2LogicalModel:ParkingRoute  
    </D2LogicalModel:parkingRoute> [0..*]  
    <D2LogicalModel:parkingColour> D2LogicalModel:RGBColour  
    </D2LogicalModel:parkingColour> [0..1] ?  
    <D2LogicalModel:onlyAssignedParking>  
    D2LogicalModel:ParkingAssignment  
    </D2LogicalModel:onlyAssignedParking> [0..1] ?  
    <D2LogicalModel:assignedParkingAmongOthers>
```

```

D2LogicalModel: ParkingAssignment
</D2LogicalModel: assignedParkingAmongOthers> [0..1] ?
<D2LogicalModel: prohibitedParking> D2LogicalModel: ParkingAssignment
</D2LogicalModel: prohibitedParking> [0..1] ?
<D2LogicalModel: tariffsAndPayment> D2LogicalModel: TariffsAndPayment
</D2LogicalModel: tariffsAndPayment> [0..1]
<D2LogicalModel: parkingEquipmentOrServiceFacility>
D2LogicalModel: ParkingRecordEquipmentOrServiceFacilityIndexParkingEquipment
</D2LogicalModel: parkingEquipmentOrServiceFacility> [0..*]
<D2LogicalModel: parkingSpace> D2LogicalModel: ParkingSpace
</D2LogicalModel: parkingSpace> [0..*] ?
<D2LogicalModel: groupOfParkingSpaces>
D2LogicalModel: GroupOfParkingSpaces
</D2LogicalModel: groupOfParkingSpaces> [0..*] ?
<D2LogicalModel: parkingThresholds> D2LogicalModel: ParkingThresholds
</D2LogicalModel: parkingThresholds> [0..1]
<D2LogicalModel: permitsAndProhibitions>
D2LogicalModel: PermitsAndProhibitions
</D2LogicalModel: permitsAndProhibitions> [0..*]
<D2LogicalModel: emergencyAssemblyPoint>
D2LogicalModel: GroupOfLocations
</D2LogicalModel: emergencyAssemblyPoint> [0..1] ?
<D2LogicalModel: entireArea> D2LogicalModel: Area
</D2LogicalModel: entireArea> [0..1] ?
<D2LogicalModel: parkingRecordDimension> D2LogicalModel: Dimension
</D2LogicalModel: parkingRecordDimension> [0..1] ?
<D2LogicalModel: parkingRecordExtension>
D2LogicalModel: ExtensionType
</D2LogicalModel: parkingRecordExtension> [0..1]
<D2LogicalModel: parkingReservation>
D2LogicalModel: ReservationTypeEnum
</D2LogicalModel: parkingReservation> [0..1] ?
<D2LogicalModel: parkingLayout> D2LogicalModel: ParkingLayoutEnum
</D2LogicalModel: parkingLayout> [0..*] ?
<D2LogicalModel: highestFloor> D2LogicalModel: Integer
</D2LogicalModel: highestFloor> [0..1] ?
<D2LogicalModel: lowestFloor> D2LogicalModel: Integer
</D2LogicalModel: lowestFloor> [0..1] ?
<D2LogicalModel: temporaryParking> D2LogicalModel: Boolean
</D2LogicalModel: temporaryParking> [0..1] ?
<D2LogicalModel: parkingSiteAddress> D2LogicalModel: Contact
</D2LogicalModel: parkingSiteAddress> [1..*] ?
<D2LogicalModel: reservationService> D2LogicalModel: Contact
</D2LogicalModel: reservationService> [0..*] ?
<D2LogicalModel: parkingUsageScenario>
D2LogicalModel: ParkingSiteScenarioIndexParkingUsageScenario
</D2LogicalModel: parkingUsageScenario> [0..*]
<D2LogicalModel: openingTimes> D2LogicalModel: OpeningTimes
</D2LogicalModel: openingTimes> [0..1]
<D2LogicalModel: parkingAccess> D2LogicalModel: ParkingAccess
</D2LogicalModel: parkingAccess> [1..*] ?
<D2LogicalModel: parkingStandardsAndSecurity>
D2LogicalModel: ParkingStandardsAndSecurity
</D2LogicalModel: parkingStandardsAndSecurity> [1]
<D2LogicalModel: parkingSiteExtension> D2LogicalModel: ExtensionType
</D2LogicalModel: parkingSiteExtension> [0..1]
<D2LogicalModel: parkingSpecialLocation>
D2LogicalModel: ParkingSpecialLocationEnum
</D2LogicalModel: parkingSpecialLocation> [1] ?
<D2LogicalModel: parkingOtherSpecialLocation>
D2LogicalModel: MultilingualString
</D2LogicalModel: parkingOtherSpecialLocation> [0..1] ?
<D2LogicalModel: specialLocationParkingSiteExtension>

```

```

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

```

Schema Component Representation

```

<xs:complexType name="SpecialLocationParkingSite">
  <xs:complexContent>
    <xs:extension base="D2LogicalModel:ParkingSite">
      <xs:sequence>
        <xs:element name="parkingSpecialLocation"
          type="D2LogicalModel:ParkingSpecialLocationEnum"
          minOccurs="1" maxOccurs="1"/>
        <xs:element name="parkingOtherSpecialLocation"
          type="D2LogicalModel:MultilingualString" minOccurs="0"
          maxOccurs="1"/>
        <xs:element name="specialLocationParkingSiteExtension"
          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
<u>Abstract</u>	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> [0..*] ?
    <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="0"
  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: **TariffsAndPayment**

Super-types: None

Sub-types: None

Name TariffsAndPayment

Abstract no

Documentation A table of charges under various conditions, primary used for parking.

XML Instance Representation

```

<...>
  <D2LogicalModel:lastUpdated> D2LogicalModel:DateTime
</D2LogicalModel:lastUpdated> [0..1] ?
  <D2LogicalModel:acceptedMeansOfPayment>
D2LogicalModel:MeansOfPaymentEnum
</D2LogicalModel:acceptedMeansOfPayment> [0..*] ?
  <D2LogicalModel:paymentMode> D2LogicalModel:ParkingPaymentModeEnum
</D2LogicalModel:paymentMode> [0..*] ?
  <D2LogicalModel:paymentAdditionalDescription>
D2LogicalModel:MultilingualString
</D2LogicalModel:paymentAdditionalDescription> [0..1] ?
  <D2LogicalModel:freeOfCharge> D2LogicalModel:Boolean
</D2LogicalModel:freeOfCharge> [0..1] ?
  <D2LogicalModel:reservationFee> D2LogicalModel:AmountOfMoney
</D2LogicalModel:reservationFee> [0..1] ?
  <D2LogicalModel:urlLinkAddress> D2LogicalModel:Url
</D2LogicalModel:urlLinkAddress> [0..1] ?
  <D2LogicalModel:chargeBand> D2LogicalModel:ChargeBand
</D2LogicalModel:chargeBand> [0..*]
  <D2LogicalModel:chargeBandByReference>
D2LogicalModel:ChargeBandByReference
</D2LogicalModel:chargeBandByReference> [0..*]
  <D2LogicalModel:acceptedPaymentCards>
D2LogicalModel:AcceptedPaymentCards
</D2LogicalModel:acceptedPaymentCards> [0..1]
  <D2LogicalModel:tariffsAndPaymentExtension>
D2LogicalModel:_ExtensionType
</D2LogicalModel:tariffsAndPaymentExtension> [0..1]
</...>

```

Schema Component Representation

```
<xs:complexType name="TariffsAndPayment">
  <xs:sequence>
    <xs:element name="lastUpdated" type="D2LogicalModel:DateTime"
      minOccurs="0" maxOccurs="1"/>
    <xs:element name="acceptedMeansOfPayment"
      type="D2LogicalModel:MeansOfPaymentEnum" minOccurs="0"
      maxOccurs="unbounded"/>
    <xs:element name="paymentMode"
      type="D2LogicalModel:ParkingPaymentModeEnum" minOccurs="0"
      maxOccurs="unbounded"/>
    <xs:element name="paymentAdditionalDescription"
      type="D2LogicalModel:MultilingualString" minOccurs="0"
      maxOccurs="1"/>
    <xs:element name="freeOfCharge" type="D2LogicalModel:Boolean"
      minOccurs="0" maxOccurs="1"/>
    <xs:element name="reservationFee"
      type="D2LogicalModel:AmountOfMoney" minOccurs="0"
      maxOccurs="1"/>
    <xs:element name="urlLinkAddress" type="D2LogicalModel:Url"
      minOccurs="0" maxOccurs="1"/>
    <xs:element name="chargeBand" type="D2LogicalModel:ChargeBand"
      minOccurs="0" maxOccurs="unbounded"/>
    <xs:element name="chargeBandByReference"
      type="D2LogicalModel:ChargeBandByReference" minOccurs="0"
      maxOccurs="unbounded"/>
    <xs:element name="acceptedPaymentCards"
      type="D2LogicalModel:AcceptedPaymentCards" minOccurs="0"/>
    <xs:element name="tariffsAndPaymentExtension"
      type="D2LogicalModel:ExtensionType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
```

[top](#)

Complex Type: **TimePeriodByHour**

Super-types: [TimePeriodOfDay](#) < TimePeriodByHour (by extension)

Sub-types: None

Name	TimePeriodByHour
<u>Abstract</u>	no
Documentation	Specification of a continuous period within a 24 hour period by times.

XML Instance Representation

```

<...>
  <D2LogicalModel:timePeriodOfDayExtension>
    D2LogicalModel: _ExtensionType
  </D2LogicalModel:timePeriodOfDayExtension> [0..1]
  <D2LogicalModel:startTimeOfPeriod> D2LogicalModel:Time
  </D2LogicalModel:startTimeOfPeriod> [1] ?
  <D2LogicalModel:endTimeOfPeriod> D2LogicalModel:Time
  </D2LogicalModel:endTimeOfPeriod> [1] ?
  <D2LogicalModel:timePeriodByHourExtension>
    D2LogicalModel: _ExtensionType
  </D2LogicalModel:timePeriodByHourExtension> [0..1]
</...>

```

Schema Component Representation

```

<xs:complexType name="TimePeriodByHour">
  <xs:complexContent>
    <xs:extension base="D2LogicalModel:TimePeriodOfDay">
      <xs:sequence>
        <xs:element name="startTimeOfPeriod"
          type="D2LogicalModel:Time" minOccurs="1" maxOccurs="1"/>
        <xs:element name="endTimeOfPeriod"
          type="D2LogicalModel:Time" minOccurs="1" maxOccurs="1"/>
        <xs:element name="timePeriodByHourExtension"
          type="D2LogicalModel: _ExtensionType" minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

```

[top](#)

Complex Type: TimePeriodOfDay

Super-types: None

Sub-types:

- [TimePeriodByHour](#) (by extension)

Name	TimePeriodOfDay
Abstract	yes
Documentation	Specification of a continuous period of time within a 24 hour period.

XML Instance Representation

```

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

```

Schema Component Representation

```

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

```

Complex Type: **TpegAreaDescriptor**

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

Sub-types: None

Name	TpegAreaDescriptor
<u>Abstract</u>	no
Documentation	A descriptor for describing an area location.

XML Instance Representation

```
<...>
  <D2LogicalModel:descriptor> D2LogicalModel:MultilingualString
</D2LogicalModel:descriptor> [1] ?
  <D2LogicalModel:tpegDescriptorExtension>
D2LogicalModel: _ExtensionType
</D2LogicalModel:tpegDescriptorExtension> [0..1]
  <D2LogicalModel:tpegAreaDescriptorType>
D2LogicalModel:TpegLoc03AreaDescriptorSubtypeEnum
</D2LogicalModel:tpegAreaDescriptorType> [1] ?
  <D2LogicalModel:tpegAreaDescriptorExtension>
D2LogicalModel: _ExtensionType
</D2LogicalModel:tpegAreaDescriptorExtension> [0..1]
</...>
```

Schema Component Representation

```
<xs:complexType name="TpegAreaDescriptor">
  <xs:complexContent>
    <xs:extension base="D2LogicalModel:TpegDescriptor">
      <xs:sequence>
        <xs:element name="tpegAreaDescriptorType"
          type="D2LogicalModel:TpegLoc03AreaDescriptorSubtypeEnum"
          minOccurs="1" maxOccurs="1"/>
        <xs:element name="tpegAreaDescriptorExtension"
          type="D2LogicalModel: _ExtensionType" minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

Complex Type: **TpegAreaLocation**

Super-types: None

Sub-types:

- [TpegGeometricArea](#) (by extension)
- [TpegNamedOnlyArea](#) (by extension)

Name	TpegAreaLocation
<u>Abstract</u>	yes

Documentation

A geographic or geometric area defined by a TPEG-Loc structure which may include height information for additional geospatial discrimination.

XML Instance Representation

```
<...>
  <D2LogicalModel:tpegAreaLocationType>
    D2LogicalModel:TpegLoc01AreaLocationSubtypeEnum
  </D2LogicalModel:tpegAreaLocationType> [1] ?
  <D2LogicalModel:tpegHeight> D2LogicalModel:TpegHeight
</D2LogicalModel:tpegHeight> [0..1]
  <D2LogicalModel:tpegAreaLocationExtension>
    D2LogicalModel:_ExtensionType
  </D2LogicalModel:tpegAreaLocationExtension> [0..1]
</...>
```

Schema Component Representation

```
<xs:complexType name="TpegAreaLocation" abstract="true">
  <xs:sequence>
    <xs:element name="tpegAreaLocationType"
      type="D2LogicalModel:TpegLoc01AreaLocationSubtypeEnum"
      minOccurs="1" maxOccurs="1"/>
    <xs:element name="tpegHeight" type="D2LogicalModel:TpegHeight"
      minOccurs="0"/>
    <xs:element name="tpegAreaLocationExtension"
      type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
```

[top](#)

Complex Type: TpegDescriptor

Super-types: None

Sub-types:

- [TpegAreaDescriptor](#) (by extension)
- [TpegPointDescriptor](#) (by extension)
 - [TpegIlocPointDescriptor](#) (by extension)
 - [TpegJunctionPointDescriptor](#) (by extension)
 - [TpegOtherPointDescriptor](#) (by extension)

Name TpegDescriptor

Abstract yes

Documentation 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="tppegDescriptorExtension"
      type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>

```

[top](#)

Complex Type: **TpegFramedPoint**

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

Sub-types: None

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

XML Instance Representation

```

<...>
  <D2LogicalModel:tppegDirection> D2LogicalModel:DirectionEnum
</D2LogicalModel:tppegDirection> [1] ?
  <D2LogicalModel:tppegPointLocationExtension>
    D2LogicalModel:_ExtensionType
  </D2LogicalModel:tppegPointLocationExtension> [0..1]
  <D2LogicalModel:tppegFramedPointLocationType>
    D2LogicalModel:TpegLoc01FramedPointLocationSubtypeEnum
  </D2LogicalModel:tppegFramedPointLocationType> [1] ?
  <D2LogicalModel:framedPoint> D2LogicalModel:TpegNonJunctionPoint
</D2LogicalModel:framedPoint> [1] ?
  <D2LogicalModel:to> D2LogicalModel:TpegPoint </D2LogicalModel:to>
  [1] ?
  <D2LogicalModel:from> D2LogicalModel:TpegPoint
</D2LogicalModel:from> [1] ?
  <D2LogicalModel:tppegFramedPointExtension>
    D2LogicalModel:_ExtensionType
  </D2LogicalModel:tppegFramedPointExtension> [0..1]
</...>

```

Schema Component Representation

```

<xs:complexType name="TpegFramedPoint">
  <xs:complexContent>
    <xs:extension base="D2LogicalModel:TpegPointLocation">
      <xs:sequence>
        <xs:element name="tppegFramedPointLocationType"
          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="tppegFramedPointExtension"
          type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

```

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

[top](#)

Complex Type: **TpegGeometricArea**

Super-types: [TpegAreaLocation](#) < **TpegGeometricArea** (by extension)
Sub-types: None

Name TpegGeometricArea
Abstract no
Documentation A geometric area defined by a centre point and a radius.

XML Instance Representation

```
<...>
  <D2LogicalModel:tpegAreaLocationType>
    D2LogicalModel:TpegLoc01AreaLocationSubTypeEnum
  </D2LogicalModel:tpegAreaLocationType> [1] ?
  <D2LogicalModel:tpegHeight> D2LogicalModel:TpegHeight
</D2LogicalModel:tpegHeight> [0..1]
  <D2LogicalModel:tpegAreaLocationExtension>
    D2LogicalModel: ExtensionType
  </D2LogicalModel:tpegAreaLocationExtension> [0..1]
  <D2LogicalModel:radius> D2LogicalModel:MetresAsNonNegativeInteger
</D2LogicalModel:radius> [1] ?
  <D2LogicalModel:centrePoint> D2LogicalModel:PointCoordinates
</D2LogicalModel:centrePoint> [1] ?
  <D2LogicalModel:name> D2LogicalModel:TpegAreaDescriptor
</D2LogicalModel:name> [0..1] ?
  <D2LogicalModel:tpegGeometricAreaExtension>
    D2LogicalModel: ExtensionType
  </D2LogicalModel:tpegGeometricAreaExtension> [0..1]
</...>
```

Schema Component Representation

```
<xs:complexType name="TpegGeometricArea">
  <xs:complexContent>
    <xs:extension base="D2LogicalModel:TpegAreaLocation">
      <xs:sequence>
        <xs:element name="radius"
          type="D2LogicalModel:MetresAsNonNegativeInteger"
          minOccurs="1" maxOccurs="1"/>
        <xs:element name="centrePoint"
          type="D2LogicalModel:PointCoordinates"/>
        <xs:element name="name"
          type="D2LogicalModel:TpegAreaDescriptor" minOccurs="0"/>
        <xs:element name="tpegGeometricAreaExtension"
          type="D2LogicalModel: ExtensionType" minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

[top](#)

Complex Type: **TpegHeight**

Super-types: None

Sub-types: None

Name TpegHeight

Abstract no

Documentation Height information which provides additional discrimination for the applicable area.

XML Instance Representation

```
<...>
  <D2LogicalModel:height> D2LogicalModel:MetresAsFloat
</D2LogicalModel:height> [0..1] ?
  <D2LogicalModel:heightType> D2LogicalModel:TpegLoc04HeightTypeEnum
</D2LogicalModel:heightType> [1] ?
  <D2LogicalModel:tpegHeightExtension> D2LogicalModel:_ExtensionType
</D2LogicalModel:tpegHeightExtension> [0..1]
</...>
```

Schema Component Representation

```
<xs:complexType name="TpegHeight">
  <xs:sequence>
    <xs:element name="height" type="D2LogicalModel:MetresAsFloat"
      minOccurs="0" maxOccurs="1"/>
    <xs:element name="heightType"
      type="D2LogicalModel:TpegLoc04HeightTypeEnum" minOccurs="1"
      maxOccurs="1"/>
    <xs:element name="tpegHeightExtension"
      type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
```

[top](#)

Complex Type: **TpegIlcPointDescriptor**

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

Sub-types: None

Name TpegIlcPointDescriptor

Abstract no

Documentation 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: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**

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

Sub-types: None

Name	TpegJunction
<u>Abstract</u>	no
Documentation	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:tpegJunctionExtension>
D2LogicalModel: _ExtensionType
</D2LogicalModel:tpegJunctionExtension> [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="tpegJunctionExtension"
          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:tpegDescriptorExtension>
    D2LogicalModel:ExtensionType
  </D2LogicalModel:tpegDescriptorExtension> [0..1]
  <D2LogicalModel:tpegPointDescriptorExtension>
    D2LogicalModel:ExtensionType
  </D2LogicalModel:tpegPointDescriptorExtension> [0..1]
  <D2LogicalModel:tpegJunctionPointDescriptorType>
    D2LogicalModel:TpegLoc03JunctionPointDescriptorSubTypeEnum
  </D2LogicalModel:tpegJunctionPointDescriptorType> [1] ?
  <D2LogicalModel:tpegJunctionPointDescriptorExtension>
    D2LogicalModel:ExtensionType
  </D2LogicalModel:tpegJunctionPointDescriptorExtension> [0..1]
</...>

```

Schema Component Representation

```

<xs:complexType name="TpegJunctionPointDescriptor">
  <xs:complexContent>
    <xs:extension base="D2LogicalModel:TpegPointDescriptor">

```

```

<xs:sequence>
  <xs:element name="tpegJunctionPointDescriptorType"
    type="D2LogicalModel:TpegLoc03JunctionPointDescriptorSubtypeEnum"
    minOccurs="1" maxOccurs="1"/>
  <xs:element name="tpegJunctionPointDescriptorExtension"
    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:tpegDirection> D2LogicalModel:DirectionEnum
</D2LogicalModel:tpegDirection> [1] ?
  <D2LogicalModel:tpegLinearLocationType>
    D2LogicalModel:TpegLoc01LinearLocationSubtypeEnum
  </D2LogicalModel:tpegLinearLocationType> [1] ?
  <D2LogicalModel:to> D2LogicalModel:TpegPoint </D2LogicalModel:to>
    [1] ?
  <D2LogicalModel:from> D2LogicalModel:TpegPoint
  </D2LogicalModel:from> [1] ?
  <D2LogicalModel:tpegLinearLocationExtension>
    D2LogicalModel:_ExtensionType
  </D2LogicalModel:tpegLinearLocationExtension> [0..1]
</...>

```

Schema Component Representation

```

<xs:complexType name="TpegLinearLocation">
  <xs:sequence>
    <xs:element name="tpegDirection"
      type="D2LogicalModel:DirectionEnum" minOccurs="1"
      maxOccurs="1"/>
    <xs:element name="tpegLinearLocationType"
      type="D2LogicalModel:TpegLoc01LinearLocationSubtypeEnum"
      minOccurs="1" maxOccurs="1"/>
    <xs:element name="to" type="D2LogicalModel:TpegPoint"/>
    <xs:element name="from" type="D2LogicalModel:TpegPoint"/>
    <xs:element name="tpegLinearLocationExtension"
      type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>

```

[top](#)

Complex Type: **TpegNamedOnlyArea**

Super-types: [TpegAreaLocation](#) < **TpegNamedOnlyArea** (by extension)

Sub-types: None

Name	TpegNamedOnlyArea
<u>Abstract</u>	no
Documentation	An area defined by a well-known name.

XML Instance Representation

```
<...>
  <D2LogicalModel:tpegAreaLocationType>
    D2LogicalModel:TpegLoc01AreaLocationSubtypeEnum
  </D2LogicalModel:tpegAreaLocationType> [1] ?
  <D2LogicalModel:tpegHeight> D2LogicalModel:TpegHeight
</D2LogicalModel:tpegHeight> [0..1]
  <D2LogicalModel:tpegAreaLocationExtension>
    D2LogicalModel:_ExtensionType
  </D2LogicalModel:tpegAreaLocationExtension> [0..1]
  <D2LogicalModel:name> D2LogicalModel:TpegAreaDescriptor
</D2LogicalModel:name> [1..*] ?
  <D2LogicalModel:tpegNamedOnlyAreaExtension>
    D2LogicalModel:_ExtensionType
  </D2LogicalModel:tpegNamedOnlyAreaExtension> [0..1]
</...>
```

Schema Component Representation

```
<xs:complexType name="TpegNamedOnlyArea">
  <xs:complexContent>
    <xs:extension base="D2LogicalModel:TpegAreaLocation">
      <xs:sequence>
        <xs:element name="name"
          type="D2LogicalModel:TpegAreaDescriptor"
          maxOccurs="unbounded"/>
        <xs:element name="tpegNamedOnlyAreaExtension"
          type="D2LogicalModel:_ExtensionType" minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

[top](#)

Complex Type: **TpegNonJunctionPoint**

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

Sub-types: None

Name	TpegNonJunctionPoint
<u>Abstract</u>	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:TpegLoc030therPointDescriptorSubTypeEnum
</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:TpegLoc030therPointDescriptorSubtypeEnum"
          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:

- [TpegIlcPointDescriptor](#) (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: **UrbanParkingSite**

Super-types: [ParkingRecord](#) < [ParkingSite](#) (by extension) < **UrbanParkingSite** (by extension)

Sub-types: None

Name UrbanParkingSite

Abstract no

Documentation A parking site in an urban context.

XML Instance Representation

```
<...
  id="xs:string [1]"
  version="xs:string [1]">
    <D2LogicalModel:parkingName> D2LogicalModel:MultilingualString
    </D2LogicalModel:parkingName> [1] ?
    <D2LogicalModel:parkingAlias> D2LogicalModel:MultilingualString
    </D2LogicalModel:parkingAlias> [0..*] ?
    <D2LogicalModel:parkingDescription>
      D2LogicalModel:MultilingualString
    </D2LogicalModel:parkingDescription> [0..1] ?
    <D2LogicalModel:parkingRecordVersionTime> D2LogicalModel:DateTime
    </D2LogicalModel:parkingRecordVersionTime> [1] ?
    <D2LogicalModel:parkingNumberOfSpaces>
      D2LogicalModel:NonNegativeInteger
    </D2LogicalModel:parkingNumberOfSpaces> [0..1] ?
    <D2LogicalModel:parkingPrincipalNumberOfSpaces>
      D2LogicalModel:NonNegativeInteger
    </D2LogicalModel:parkingPrincipalNumberOfSpaces> [0..1] ?
    <D2LogicalModel:maximumParkingDuration> D2LogicalModel:Seconds
    </D2LogicalModel:maximumParkingDuration> [0..1] ?
    <D2LogicalModel:photoUrl> D2LogicalModel:Url
    </D2LogicalModel:photoUrl> [0..1] ?
    <D2LogicalModel:urlLinkAddress> D2LogicalModel:Url
    </D2LogicalModel:urlLinkAddress> [0..1] ?
    <D2LogicalModel:parkingOccupancyDetectionType>
      D2LogicalModel:OccupancyDetectionTypeEnum
    </D2LogicalModel:parkingOccupancyDetectionType> [0..*] ?
    <D2LogicalModel:emergencyContact> D2LogicalModel:Contact
    </D2LogicalModel:emergencyContact> [0..*] ?
    <D2LogicalModel:owner> D2LogicalModel:Contact
    </D2LogicalModel:owner> [0..*] ?
    <D2LogicalModel:responisbleAuthority> D2LogicalModel:Contact
    </D2LogicalModel:responisbleAuthority> [0..*] ?
    <D2LogicalModel:securityService> D2LogicalModel:Contact
    </D2LogicalModel:securityService> [0..*] ?
    <D2LogicalModel:operator> D2LogicalModel:Contact
    </D2LogicalModel:operator> [1..*] ?
    <D2LogicalModel:servicePartner> D2LogicalModel:Contact
    </D2LogicalModel:servicePartner> [0..*] ?
    <D2LogicalModel:parkingVMS> D2LogicalModel:ParkingVMS
    </D2LogicalModel:parkingVMS> [0..*]
    <D2LogicalModel:parkingLocation> D2LogicalModel:GroupOfLocations
    </D2LogicalModel:parkingLocation> [1] ?
    <D2LogicalModel:parkingRoute> D2LogicalModel:ParkingRoute
    </D2LogicalModel:parkingRoute> [0..*]
    <D2LogicalModel:parkingColour> D2LogicalModel:RGBColour
    </D2LogicalModel:parkingColour> [0..1] ?
    <D2LogicalModel:onlyAssignedParking>
      D2LogicalModel:ParkingAssignment
```

```

</D2LogicalModel:onlyAssignedParking> [0..1] ?
<D2LogicalModel:assignedParkingAmongOthers>
D2LogicalModel: ParkingAssignment
</D2LogicalModel:assignedParkingAmongOthers> [0..1] ?
<D2LogicalModel:prohibitedParking> D2LogicalModel: ParkingAssignment
</D2LogicalModel:prohibitedParking> [0..1] ?
<D2LogicalModel:tariffsAndPayment> D2LogicalModel: TariffsAndPayment
</D2LogicalModel:tariffsAndPayment> [0..1]
<D2LogicalModel:parkingEquipmentOrServiceFacility>
D2LogicalModel: ParkingRecordEquipmentOrServiceFacilityIndexParkingEquipment
</D2LogicalModel:parkingEquipmentOrServiceFacility> [0..*]
<D2LogicalModel:parkingSpace> D2LogicalModel: ParkingSpace
</D2LogicalModel:parkingSpace> [0..*] ?
<D2LogicalModel:groupOfParkingSpaces>
D2LogicalModel: GroupOfParkingSpaces
</D2LogicalModel:groupOfParkingSpaces> [0..*] ?
<D2LogicalModel:parkingThresholds> D2LogicalModel: ParkingThresholds
</D2LogicalModel:parkingThresholds> [0..1]
<D2LogicalModel:permitsAndProhibitions>
D2LogicalModel: PermitsAndProhibitions
</D2LogicalModel:permitsAndProhibitions> [0..*]
<D2LogicalModel:emergencyAssemblyPoint>
D2LogicalModel: GroupOfLocations
</D2LogicalModel:emergencyAssemblyPoint> [0..1] ?
<D2LogicalModel:entireArea> D2LogicalModel: Area
</D2LogicalModel:entireArea> [0..1] ?
<D2LogicalModel:parkingRecordDimension> D2LogicalModel: Dimension
</D2LogicalModel:parkingRecordDimension> [0..1] ?
<D2LogicalModel:parkingRecordExtension>
D2LogicalModel: ExtensionType
</D2LogicalModel:parkingRecordExtension> [0..1]
<D2LogicalModel:parkingReservation>
D2LogicalModel: ReservationTypeEnum
</D2LogicalModel:parkingReservation> [0..1] ?
<D2LogicalModel:parkingLayout> D2LogicalModel: ParkingLayoutEnum
</D2LogicalModel:parkingLayout> [0..*] ?
<D2LogicalModel:highestFloor> D2LogicalModel: Integer
</D2LogicalModel:highestFloor> [0..1] ?
<D2LogicalModel:lowestFloor> D2LogicalModel: Integer
</D2LogicalModel:lowestFloor> [0..1] ?
<D2LogicalModel:temporaryParking> D2LogicalModel: Boolean
</D2LogicalModel:temporaryParking> [0..1] ?
<D2LogicalModel:parkingSiteAddress> D2LogicalModel: Contact
</D2LogicalModel:parkingSiteAddress> [1..*] ?
<D2LogicalModel:reservationService> D2LogicalModel: Contact
</D2LogicalModel:reservationService> [0..*] ?
<D2LogicalModel:parkingUsageScenario>
D2LogicalModel: ParkingSiteScenarioIndexParkingUsageScenario
</D2LogicalModel:parkingUsageScenario> [0..*]
<D2LogicalModel:openingTimes> D2LogicalModel: OpeningTimes
</D2LogicalModel:openingTimes> [0..1]
<D2LogicalModel:parkingAccess> D2LogicalModel: ParkingAccess
</D2LogicalModel:parkingAccess> [1..*] ?
<D2LogicalModel:parkingStandardsAndSecurity>
D2LogicalModel: ParkingStandardsAndSecurity
</D2LogicalModel:parkingStandardsAndSecurity> [1]
<D2LogicalModel:parkingSiteExtension> D2LogicalModel: ExtensionType
</D2LogicalModel:parkingSiteExtension> [0..1]
<D2LogicalModel:urbanParkingSiteType>
D2LogicalModel: UrbanParkingSiteTypeEnum
</D2LogicalModel:urbanParkingSiteType> [1] ?
<D2LogicalModel:parkingZone> D2LogicalModel: MultilingualString

```

```

</D2LogicalModel:parkingZone> [0..1] ?
<D2LogicalModel:urbanParkingSiteExtension>
D2LogicalModel: _ExtensionType
</D2LogicalModel:urbanParkingSiteExtension> [0..1]
</...>

```

Schema Component Representation

```

<xs:complexType name="UrbanParkingSite">
  <xs:complexContent>
    <xs:extension base="D2LogicalModel:ParkingSite">
      <xs:sequence>
        <xs:element name="urbanParkingSiteType"
          type="D2LogicalModel:UrbanParkingSiteTypeEnum"
          minOccurs="1" maxOccurs="1"/>
        <xs:element name="parkingZone"
          type="D2LogicalModel:MultilingualString" minOccurs="0"
          maxOccurs="1"/>
        <xs:element name="urbanParkingSiteExtension"
          type="D2LogicalModel: _ExtensionType" minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

```

[top](#)

Complex Type: **Validity**

Super-types: None

Sub-types: None

Name	Validity
<u>Abstract</u>	no
Documentation	Specification of validity, either explicitly or by a validity time period specification which may be discontinuous.

XML Instance Representation

```

<...>
  <D2LogicalModel:validityStatus> D2LogicalModel:ValidityStatusEnum
</D2LogicalModel:validityStatus> [1] ?
  <D2LogicalModel:overrunning> D2LogicalModel:Boolean
</D2LogicalModel:overrunning> [0..1] ?
  <D2LogicalModel:validityTimeSpecification>
D2LogicalModel:OverallPeriod
</D2LogicalModel:validityTimeSpecification> [1] ?
  <D2LogicalModel:validityExtension> D2LogicalModel: _ExtensionType
</D2LogicalModel:validityExtension> [0..1]
</...>

```

Schema Component Representation

```

<xs:complexType name="Validity">
  <xs:sequence>
    <xs:element name="validityStatus"
      type="D2LogicalModel:ValidityStatusEnum" minOccurs="1"
      maxOccurs="1"/>
    <xs:element name="overrunning" type="D2LogicalModel:Boolean"

```

```

        minOccurs="0" maxOccurs="1"/>
        <xs:element name="validityTimeSpecification"
        type="D2LogicalModel:OverallPeriod"/>
        <xs:element name="validityExtension"
        type="D2LogicalModel:ExtensionType" minOccurs="0"/>
    </xs:sequence>
</xs:complexType>

```

[top](#)

Complex Type: **VehicleCharacteristics**

Super-types: None

Sub-types: None

Name	VehicleCharacteristics
Abstract	no
Documentation	The characteristics of a vehicle, e.g. lorry of gross weight greater than 30 tonnes.

XML Instance Representation

```

<...>
  <D2LogicalModel:fuelType> D2LogicalModel:FuelTypeEnum
</D2LogicalModel:fuelType> [0..1] ?
  <D2LogicalModel:loadType> D2LogicalModel:LoadTypeEnum
</D2LogicalModel:loadType> [0..1] ?
  <D2LogicalModel:vehicleEquipment>
    D2LogicalModel:VehicleEquipmentEnum
  </D2LogicalModel:vehicleEquipment> [0..1] ?
  <D2LogicalModel:vehicleType> D2LogicalModel:VehicleTypeEnum
</D2LogicalModel:vehicleType> [0..*] ?
  <D2LogicalModel:grossWeightCharacteristic>
    D2LogicalModel:GrossWeightCharacteristic
  </D2LogicalModel:grossWeightCharacteristic> [0..2]
  <D2LogicalModel:heightCharacteristic>
    D2LogicalModel:HeightCharacteristic
  </D2LogicalModel:heightCharacteristic> [0..2]
  <D2LogicalModel:lengthCharacteristic>
    D2LogicalModel:LengthCharacteristic
  </D2LogicalModel:lengthCharacteristic> [0..2]
  <D2LogicalModel:widthCharacteristic>
    D2LogicalModel:WidthCharacteristic
  </D2LogicalModel:widthCharacteristic> [0..2]
  <D2LogicalModel:heaviestAxleWeightCharacteristic>
    D2LogicalModel:HeaviestAxleWeightCharacteristic
  </D2LogicalModel:heaviestAxleWeightCharacteristic> [0..2]
  <D2LogicalModel:numberOfAxlesCharacteristic>
    D2LogicalModel:NumberOfAxlesCharacteristic
  </D2LogicalModel:numberOfAxlesCharacteristic> [0..2]
  <D2LogicalModel:vehicleCharacteristicsExtension>
    D2LogicalModel:VehicleCharacteristicsExtensionType
  </D2LogicalModel:vehicleCharacteristicsExtension> [0..1]
</...>

```

Schema Component Representation

```

<xs:complexType name="VehicleCharacteristics">
  <xs:sequence>

```

```

<xs:element name="fuelType" type="D2LogicalModel:FuelTypeEnum"
minOccurs="0" maxOccurs="1"/>
<xs:element name="loadType" type="D2LogicalModel:LoadTypeEnum"
minOccurs="0" maxOccurs="1"/>
<xs:element name="vehicleEquipment"
type="D2LogicalModel:VehicleEquipmentEnum" minOccurs="0"
maxOccurs="1"/>
<xs:element name="vehicleType"
type="D2LogicalModel:VehicleTypeEnum" minOccurs="0"
maxOccurs="unbounded"/>
<xs:element name="grossWeightCharacteristic"
type="D2LogicalModel:GrossWeightCharacteristic" minOccurs="0"
maxOccurs="2"/>
<xs:element name="heightCharacteristic"
type="D2LogicalModel:HeightCharacteristic" minOccurs="0"
maxOccurs="2"/>
<xs:element name="lengthCharacteristic"
type="D2LogicalModel:LengthCharacteristic" minOccurs="0"
maxOccurs="2"/>
<xs:element name="widthCharacteristic"
type="D2LogicalModel:WidthCharacteristic" minOccurs="0"
maxOccurs="2"/>
<xs:element name="heaviestAxleWeightCharacteristic"
type="D2LogicalModel:HeaviestAxleWeightCharacteristic"
minOccurs="0" maxOccurs="2"/>
<xs:element name="numberOfAxlesCharacteristic"
type="D2LogicalModel:NumberOfAxlesCharacteristic" minOccurs="0"
maxOccurs="2"/>
<xs:element name="vehicleCharacteristicsExtension"
type="D2LogicalModel:VehicleCharacteristicsExtensionType"
minOccurs="0"/>
</xs:sequence>
</xs:complexType>

```

[top](#)

Complex Type: **VehicleCharacteristicsExtended**

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

Name	VehicleCharacteristicsExtended
Abstract	no
Documentation	Extension point for 'VehicleCharacteristics' to support additional attributes and literals like additional fuel types, load types etc.

XML Instance Representation

```

<...>
  <D2LogicalModel:loadType2> D2LogicalModel:LoadType2Enum
</D2LogicalModel:loadType2> [0..1] ?
  <D2LogicalModel:vehicleType2> D2LogicalModel:VehicleType2Enum
</D2LogicalModel:vehicleType2> [0..1] ?
</...>

```

Schema Component Representation

```

<xs:complexType name="VehicleCharacteristicsExtended">
  <xs:sequence>
    <xs:element name="loadType2" type="D2LogicalModel:LoadType2Enum"

```

```

minOccurs="0" maxOccurs="1"/>
<xs:element name="vehicleType2"
  type="D2LogicalModel:VehicleType2Enum" minOccurs="0"
  maxOccurs="1"/>
</xs:sequence>
</xs:complexType>

```

[top](#)

Complex Type: **VersionedReference**

Super-types: None

Sub-types:

- [ChargeBandVersionedReference](#) (by extension)
- [ContactDetailsVersionedReference](#) (by extension)
- [ParkingRecordVersionedReference](#) (by extension)
- [ParkingRouteDetailsVersionedReference](#) (by extension)
- [PredefinedItineraryVersionedReference](#) (by extension)
- [PredefinedLocationVersionedReference](#) (by extension)
- [PredefinedNonOrderedLocationGroupVersionedReference](#) (by extension)
- [VmsUnitRecordVersionedReference](#) (by extension)

Name VersionedReference

Abstract 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: **WidthCharacteristic**

Super-types: None

Sub-types: None

Name WidthCharacteristic

Abstract no

Documentation Width characteristic of a vehicle.

XML Instance Representation

```

<...>
  <D2LogicalModel:comparisonOperator>
    D2LogicalModel:ComparisonOperatorEnum
  </D2LogicalModel:comparisonOperator> [1] ?

```

```

<D2LogicalModel:vehicleWidth> D2LogicalModel:MetresAsFloat
</D2LogicalModel:vehicleWidth> [1] ?
<D2LogicalModel:widthCharacteristicExtension>
D2LogicalModel: _ExtensionType
</D2LogicalModel:widthCharacteristicExtension> [0..1]
</...>

```

Schema Component Representation

```

<xs:complexType name="WidthCharacteristic">
  <xs:sequence>
    <xs:element name="comparisonOperator"
      type="D2LogicalModel:ComparisonOperatorEnum" minOccurs="1"
      maxOccurs="1"/>
    <xs:element name="vehicleWidth"
      type="D2LogicalModel:MetresAsFloat" minOccurs="1"
      maxOccurs="1"/>
    <xs:element name="widthCharacteristicExtension"
      type="D2LogicalModel: _ExtensionType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>

```

[top](#)

Complex Type: **_AreaExtensionType**

Super-types: None

Sub-types: None

Name _AreaExtensionType

Abstract no

XML Instance Representation

```

<...>
  <D2LogicalModel:areaExtended> D2LogicalModel:AreaExtended
</D2LogicalModel:areaExtended> [0..1]
  Allow any elements from a namespace other than this schema's
  namespace (lax validation). [0..*]
</...>

```

Schema Component Representation

```

<xs:complexType name="_AreaExtensionType">
  <xs:sequence>
    <xs:element name="areaExtended"
      type="D2LogicalModel:AreaExtended" minOccurs="0"/>
    <xs:any namespace="##other" processContents="lax" minOccurs="0"
      maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>

```

[top](#)

Complex Type: **_ChargeBandVersionedReference**

Super-types: [VersionedReference](#) < _ChargeBandVersionedReference (by extension)

Sub-types: None

Name _ChargeBandVersionedReference

Abstract no

XML Instance Representation

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

Schema Component Representation

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

[top](#)

Complex Type: _ContactDetailsVersionedReference

Super-types: [VersionedReference](#) < _ContactDetailsVersionedReference (by extension)

Sub-types: None

Name _ContactDetailsVersionedReference

Abstract no

XML Instance Representation

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

Schema Component Representation

```
<xs:complexType name="_ContactDetailsVersionedReference">  
  <xs:complexContent>  
    <xs:extension base="D2LogicalModel:VersionedReference">  
      <xs:attribute name="targetClass" use="required"  
        fixed="ContactDetails"/>  
    </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: **_GenericPublicationExtensionType**

Super-types: None

Sub-types: None

Name _GenericPublicationExtensionType

Abstract no

XML Instance Representation

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

Schema Component Representation

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

[top](#)

Complex Type: **_GroupOfParkingSpaces**

Super-types: None

Sub-types: None

Name _GroupOfParkingSpaces

Abstract no

XML Instance Representation

```
<...  
  groupIndex="xs:int [1]">  
    <D2LogicalModel:parkingSpaceBasics>  
      D2LogicalModel:ParkingSpaceBasics  
    </D2LogicalModel:parkingSpaceBasics> [1]  
</...>
```

Schema Component Representation

```
<xs:complexType name="_GroupOfParkingSpaces">  
  <xs:sequence>  
    <xs:element name="parkingSpaceBasics"  
      type="D2LogicalModel:ParkingSpaceBasics" minOccurs="1"  
      maxOccurs="1"/>  
  </xs:sequence>  
  <xs:attribute name="groupIndex" type="xs:int" use="required"/>  
</xs:complexType>
```

[top](#)

Complex Type: _GroupOfParkingSpacesParkingSpaceIndexParkingSpace

Super-types: None

Sub-types: None

Name _GroupOfParkingSpacesParkingSpaceIndexParkingSpace

Abstract no

XML Instance Representation

```
<...  
  parkingSpaceIndex="xs:int [1]">  
    <D2LogicalModel:parkingSpace> D2LogicalModel:ParkingSpace  
  </D2LogicalModel:parkingSpace> [1]  
</...>
```

Schema Component Representation

```
<xs:complexType  
name="_GroupOfParkingSpacesParkingSpaceIndexParkingSpace">  
  <xs:sequence>  
    <xs:element name="parkingSpace"  
      type="D2LogicalModel:ParkingSpace" minOccurs="1" maxOccurs="1"/>  
  </xs:sequence>  
  <xs:attribute name="parkingSpaceIndex" type="xs:int"  
    use="required"/>  
</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: LocationContainedInItinerary

Super-types: None

Sub-types: None

Name LocationContainedInItinerary

Abstract no

XML Instance Representation

```
<...  
  index="xs:int [1]">  
    <D2LogicalModel:location> D2LogicalModel:Location  
  </D2LogicalModel:location> [1]  
</...>
```

Schema Component Representation

```
<xs:complexType name="_LocationContainedInItinerary">  
  <xs:sequence>  
    <xs:element name="location" type="D2LogicalModel:Location"  
      minOccurs="1" maxOccurs="1"/>  
  </xs:sequence>  
  <xs:attribute name="index" type="xs:int" use="required"/>  
</xs:complexType>
```

[top](#)

Complex Type: **_ParkingAccessReference**

Super-types: [Reference](#) < **_ParkingAccessReference** (by extension)

Sub-types: None

Name **_ParkingAccessReference**

Abstract no

XML Instance Representation

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

Schema Component Representation

```
<xs:complexType name="_ParkingAccessReference">  
  <xs:complexContent>  
    <xs:extension base="D2LogicalModel:Reference">  
      <xs:attribute name="targetClass" use="required"  
        fixed="ParkingAccess"/>  
    </xs:extension>  
  </xs:complexContent>  
</xs:complexType>
```

[top](#)

Complex Type:

_ParkingRecordEquipmentOrServiceFacilityIndexParkingEquipmentOrServiceFacilit

Super-types: None

Sub-types: None

Name **_ParkingRecordEquipmentOrServiceFacilityIndexParkingEquipmentOrServiceFacility**

Abstract no

XML Instance Representation

```
<...  
  equipmentOrServiceFacilityIndex="xs:int [1]">  
    <D2LogicalModel:parkingEquipmentOrServiceFacility>  
      D2LogicalModel:ParkingEquipmentOrServiceFacility  
    </D2LogicalModel:parkingEquipmentOrServiceFacility> [1]  
</...>
```

Schema Component Representation

```
<xs:complexType  
  name="_ParkingRecordEquipmentOrServiceFacilityIndexParkingEquipmentOrServiceFa"  
  <xs:sequence>  
    <xs:element name="parkingEquipmentOrServiceFacility"  
      type="D2LogicalModel:ParkingEquipmentOrServiceFacility"  
      minOccurs="1" maxOccurs="1"/>  
    </xs:sequence>  
    <xs:attribute name="equipmentOrServiceFacilityIndex" type="xs:int"  
      use="required"/>  
  </xs:complexType>
```

Complex Type: **_ParkingRecordVersionedReference**

Super-types: [VersionedReference](#) < **_ParkingRecordVersionedReference** (by extension)

Sub-types: None

Name **_ParkingRecordVersionedReference**

Abstract no

XML Instance Representation

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

Schema Component Representation

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

Complex Type: **_ParkingRouteDetailsVersionedReference**

Super-types: [VersionedReference](#) < **_ParkingRouteDetailsVersionedReference** (by extension)

Sub-types: None

Name **_ParkingRouteDetailsVersionedReference**

Abstract no

XML Instance Representation

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

Schema Component Representation

```

<xs:complexType name="_ParkingRouteDetailsVersionedReference">
  <xs:complexContent>
    <xs:extension base="D2LogicalModel:VersionedReference">
      <xs:attribute name="targetClass" use="required"
        fixed="ParkingRouteDetails"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

```

[top](#)

Complex Type: **_ParkingSiteScenarioIndexParkingUsageScenario**

Super-types: None

Sub-types: None

Name _ParkingSiteScenarioIndexParkingUsageScenario

Abstract no

XML Instance Representation

```

<...
  scenarioIndex="xs:int [1]">
    <D2LogicalModel:parkingUsageScenario>
      D2LogicalModel:ParkingUsageScenario
    </D2LogicalModel:parkingUsageScenario> [1]
  </...>

```

Schema Component Representation

```

<xs:complexType name="_ParkingSiteScenarioIndexParkingUsageScenario">
  <xs:sequence>
    <xs:element name="parkingUsageScenario"
      type="D2LogicalModel:ParkingUsageScenario" minOccurs="1"
      maxOccurs="1"/>
  </xs:sequence>
  <xs:attribute name="scenarioIndex" type="xs:int" use="required"/>
</xs:complexType>

```

[top](#)

Complex Type: **_ParkingSpace**

Super-types: None

Sub-types: None

Name _ParkingSpace

Abstract no

XML Instance Representation

```

<...
  parkingSpaceIndex="xs:int [1]">
    <D2LogicalModel:parkingSpaceBasics>
      D2LogicalModel:ParkingSpaceBasics
    </D2LogicalModel:parkingSpaceBasics> [1]
  </...>

```

</...>

Schema Component Representation

```
<xs:complexType name="_ParkingSpace">
  <xs:sequence>
    <xs:element name="parkingSpaceBasics"
      type="D2LogicalModel:ParkingSpaceBasics" minOccurs="1"
      maxOccurs="1"/>
  </xs:sequence>
  <xs:attribute name="parkingSpaceIndex" type="xs:int"
    use="required"/>
</xs:complexType>
```

[top](#)

Complex Type:

_ParkingSpaceBasicsEquipmentOrServiceFacilityIndexParkingEquipmentOrServiceFacility

Super-types: None

Sub-types: None

Name _ParkingSpaceBasicsEquipmentOrServiceFacilityIndexParkingEquipmentOrServiceFacility

Abstract no

XML Instance Representation

```
<...
  equipmentOrServiceFacilityIndex="xs:int [1]">
    <D2LogicalModel:parkingEquipmentOrServiceFacility>
      D2LogicalModel:ParkingEquipmentOrServiceFacility
    </D2LogicalModel:parkingEquipmentOrServiceFacility> [1]
  </...>
```

Schema Component Representation

```
<xs:complexType
  name="_ParkingSpaceBasicsEquipmentOrServiceFacilityIndexParkingEquipmentOrServ
  <xs:sequence>
    <xs:element name="parkingEquipmentOrServiceFacility"
      type="D2LogicalModel:ParkingEquipmentOrServiceFacility"
      minOccurs="1" maxOccurs="1"/>
  </xs:sequence>
  <xs:attribute name="equipmentOrServiceFacilityIndex" type="xs:int"
    use="required"/>
</xs:complexType>
```

[top](#)

Complex Type: **_ParkingSpaceBasicsScenarioIndexParkingUsageScenario**

Super-types: None

Sub-types: None

Name _ParkingSpaceBasicsScenarioIndexParkingUsageScenario

Abstract no

XML Instance Representation

```
<...  
  scenarioIndex="xs:int [1]">  
    <D2LogicalModel:parkingUsageScenario>  
      D2LogicalModel:ParkingUsageScenario  
    </D2LogicalModel:parkingUsageScenario> [1]  
</...>
```

Schema Component Representation

```
<xs:complexType  
  name="_ParkingSpaceBasicsScenarioIndexParkingUsageScenario">  
  <xs:sequence>  
    <xs:element name="parkingUsageScenario"  
      type="D2LogicalModel:ParkingUsageScenario" minOccurs="1"  
      maxOccurs="1"/>  
  </xs:sequence>  
  <xs:attribute name="scenarioIndex" type="xs:int" use="required"/>  
</xs:complexType>
```

[top](#)

Complex Type: **_PeriodExtensionType**

Super-types: None

Sub-types: None

Name _PeriodExtensionType

Abstract no

XML Instance Representation

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

Schema Component Representation

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

[top](#)

Complex Type: PolygonAreaIndexPointCoordinates

Super-types: None

Sub-types: None

Name PolygonAreaIndexPointCoordinates

Abstract no

XML Instance Representation

```
<...
  index="xs:int [1]">
    <D2LogicalModel:pointCoordinates> D2LogicalModel:PointCoordinates
  </D2LogicalModel:pointCoordinates> [1]
</...>
```

Schema Component Representation

```
<xs:complexType name="_PolygonAreaIndexPointCoordinates">
  <xs:sequence>
    <xs:element name="pointCoordinates"
      type="D2LogicalModel:PointCoordinates" minOccurs="1"
      maxOccurs="1"/>
  </xs:sequence>
  <xs:attribute name="index" type="xs:int" use="required"/>
</xs:complexType>
```

[top](#)

Complex Type: PredefinedItineraryVersionedReference

Super-types: [VersionedReference](#) < PredefinedItineraryVersionedReference (by

	extension)
Sub-types:	None

Name `_PredefinedItineraryVersionedReference`

Abstract no

XML Instance Representation

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

Schema Component Representation

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

[top](#)

Complex Type: `_PredefinedLocationVersionedReference`

Super-types:	VersionedReference < <code>_PredefinedLocationVersionedReference</code> (by extension)
--------------	--

Sub-types:	None
------------	------

Name `_PredefinedLocationVersionedReference`

Abstract no

XML Instance Representation

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

Schema Component Representation

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

[top](#)

Complex Type: `_PredefinedNonOrderedLocationGroupVersionedReference`

Super-types: [VersionedReference](#) <
 _PredefinedNonOrderedLocationGroupVersionedReference (by
 extension)
 Sub-types: None

Name _PredefinedNonOrderedLocationGroupVersionedReference
Abstract no

XML Instance Representation

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

Schema Component Representation

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

[top](#)

Complex Type: _VehicleCharacteristicsExtensionType

Super-types: None
 Sub-types: None

Name _VehicleCharacteristicsExtensionType
Abstract no

XML Instance Representation

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

Schema Component Representation

```

<xs:complexType name="_VehicleCharacteristicsExtensionType">
  <xs:sequence>
    <xs:element name="vehicleCharacteristicsExtended"
      type="D2LogicalModel:VehicleCharacteristicsExtended"
      minOccurs="0"/>
    <xs:any namespace="##other" processContents="lax" minOccurs="0"
      maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>

```

[top](#)

Complex Type: **_VmsUnitRecordVersionedReference**

Super-types: [VersionedReference](#) < **_VmsUnitRecordVersionedReference** (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](#)

Simple Type: **AccessCategoryEnum**

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

Sub-types: None

Name **AccessCategoryEnum**

Content

- Base XSD Type: string
- *value* comes from list:
{'vehicleEntranceAndExit'|'vehicleEntrance'|'vehicleExit'|'pedestrianEntrance'.

Documentation Specifies the category of the access.

Schema Component Representation

```

<xs:simpleType name="AccessCategoryEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="vehicleEntranceAndExit"/>
    <xs:enumeration value="vehicleEntrance"/>
    <xs:enumeration value="vehicleExit"/>
    <xs:enumeration value="pedestrianEntranceAndExit"/>
    <xs:enumeration value="pedestrianEntrance"/>
    <xs:enumeration value="pedestrianExit"/>
    <xs:enumeration value="emergencyExit"/>
    <xs:enumeration value="unspecified"/>
    <xs:enumeration value="unknown"/>
    <xs:enumeration value="other"/>
  </xs:restriction>
</xs:simpleType>

```

[top](#)

Simple Type: **AccessEquipmentEnum**

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

Sub-types: None

Name AccessEquipmentEnum

Content

- Base XSD Type: string
- *value* comes from list:
{'barrier'|'trafficSignal'|'ticketButtonMachine'|'ticketCardMachine'|'payAndExit|

Documentation Specifies additional equipment for this access.

Schema Component Representation

```

<xs:simpleType name="AccessEquipmentEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="barrier"/>
    <xs:enumeration value="trafficSignal"/>
    <xs:enumeration value="ticketButtonMachine"/>
    <xs:enumeration value="ticketCardMachine"/>
    <xs:enumeration value="payAndExitMachine"/>
    <xs:enumeration value="other"/>
  </xs:restriction>
</xs:simpleType>

```

[top](#)

Simple Type: **AccessibilityEnum**

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

Sub-types: None

Name AccessibilityEnum

Content

- Base XSD Type: string
- *value* comes from list:

{'barrierFreeAccessible'|'handicappedAccessible'|'wheelChairAccessible'|'ha

Documentation Special forms of accessibility, easements and marking for handicapped people.

Schema Component Representation

```
<xs:simpleType name="AccessibilityEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="barrierFreeAccessible"/>
    <xs:enumeration value="handicappedAccessible"/>
    <xs:enumeration value="wheelChairAccessible"/>
    <xs:enumeration value="handicappedEasements"/>
    <xs:enumeration value="orientationSystemForBlindPeople"/>
    <xs:enumeration value="handicappedMarked"/>
    <xs:enumeration value="none"/>
    <xs:enumeration value="unknown"/>
    <xs:enumeration value="other"/>
  </xs:restriction>
</xs:simpleType>
```

[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: AmountOfMoney

Super-types: [xs:decimal](#) < [Decimal](#) (by restriction) < **AmountOfMoney** (by restriction)

Sub-types: None

Name	AmountOfMoney
Content	• Base XSD Type: decimal • <i>total no. of digits</i> = 8 • <i>no. of fraction digits</i> = 2
Documentation	A monetary value expressed to two decimal places.

Schema Component Representation

```
<xs:simpleType name="AmountOfMoney">
  <xs:restriction base="D2LogicalModel:Decimal">
    <xs:totalDigits value="8"/>
    <xs:fractionDigits value="2"/>
  </xs:restriction>
</xs:simpleType>
```

[top](#)

Simple Type: Ampere

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

Sub-types: None

Name	Ampere
Content	• Base XSD Type: float
Documentation	Ampere.

Schema Component Representation

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

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

Simple Type: **AvailabilityEnum**

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

Sub-types: None

Name AvailabilityEnum

Content

- Base XSD Type: string
- *value* comes from list:
{'available'|'notAvailable'|'unknown'}

Documentation An enumeration which states if something is available or not.

Schema Component Representation

```
<xs:simpleType name="AvailabilityEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="available"/>
    <xs:enumeration value="notAvailable"/>
    <xs:enumeration value="unknown"/>
  </xs:restriction>
</xs:simpleType>
```

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'|'leftHandFee

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: **ChargeTypeEnum**

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

Sub-types: None

Name ChargeTypeEnum

Content

- Base XSD Type: string
- *value* comes from list:
{'minimum'|'maximum'|'additionalIntervalPrice'|'seasonTicket'|'temporaryPrice'}

Documentation Charge type

Schema Component Representation

```
<xs:simpleType name="ChargeTypeEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="minimum"/>
    <xs:enumeration value="maximum"/>
    <xs:enumeration value="additionalIntervalPrice"/>
    <xs:enumeration value="seasonTicket"/>
    <xs:enumeration value="temporaryPrice"/>
    <xs:enumeration value="firstIntervalPrice"/>
    <xs:enumeration value="freeParking"/>
    <xs:enumeration value="flat"/>
    <xs:enumeration value="unknown"/>
    <xs:enumeration value="other"/>
  </xs:restriction>
</xs:simpleType>
```

[top](#)

Simple Type: **ChargingStationUsageTypeEnum**

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

Sub-types: None

Name ChargingStationUsageTypeEnum

Content

- Base XSD Type: string
- *value* comes from list:
{'electricVehicle'|'motorhomeOrCaravanSupply'|'electricBikeOrMotorcycle'|'lo'}

Documentation Type of usage for electric charging station(s).

Schema Component Representation

```
<xs:simpleType name="ChargingStationUsageTypeEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="electricVehicle"/>
    <xs:enumeration value="motorhomeOrCaravanSupply"/>
    <xs:enumeration value="electricBikeOrMotorcycle"/>
    <xs:enumeration value="lorryPowerConsumption"/>
    <xs:enumeration value="electricalDevices"/>
    <xs:enumeration value="other"/>
  </xs:restriction>
</xs:simpleType>
```

[top](#)

Simple Type: **ComparisonOperatorEnum**

<i>Super-types:</i>	xs:string < ComparisonOperatorEnum (by restriction)
<i>Sub-types:</i>	None

Name ComparisonOperatorEnum

Content

- Base XSD Type: string
- *value* comes from list:
{'equalTo'|'greaterThan'|'greaterThanOrEqualTo'|'lessThan'|'lessThanOrEqualTo'}

Documentation Logical comparison operations.

Schema Component Representation

```
<xs:simpleType name="ComparisonOperatorEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="equalTo"/>
    <xs:enumeration value="greaterThan"/>
    <xs:enumeration value="greaterThanOrEqualTo"/>
    <xs:enumeration value="lessThan"/>
    <xs:enumeration value="lessThanOrEqualTo"/>
  </xs:restriction>
</xs:simpleType>
```

[top](#)

Simple Type: **ConfidentialityValueEnum**

<i>Super-types:</i>	xs:string < ConfidentialityValueEnum (by restriction)
<i>Sub-types:</i>	None

Name ConfidentialityValueEnum

Content

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

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

Simple Type: **CountryEnum**

Super-types: [xs:string](#) < **CountryEnum** (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'}

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

```

<xs:enumeration value="sm"/>
<xs:enumeration value="tr"/>
<xs:enumeration value="va"/>
<xs:enumeration value="other"/>
</xs:restriction>
</xs:simpleType>

```

[top](#)

Simple Type: **CubicMetres**

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

Name CubicMetres

Content

- Base XSD Type: float

Documentation A volumetric measure defined in cubic metres.

Schema Component Representation

```

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

```

[top](#)

Simple Type: **CurrencyEnum**

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

Name CurrencyEnum

Content

- Base XSD Type: string
- *value* comes from list:
 { 'eur' | 'all' | 'amd' | 'azn' | 'bam' | 'bgn' | 'byr' | 'chf' | 'czk' | 'dkk' | 'gbp' | 'gel' | 'hrk' | 'huf' | 'isk' | 'ltl' }

Documentation Three letter code defining the currency according to ISO 4217 (e.g. EUR for Euro). The US dollar.

Schema Component Representation

```

<xs:simpleType name="CurrencyEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="eur"/>
    <xs:enumeration value="all"/>
    <xs:enumeration value="amd"/>
    <xs:enumeration value="azn"/>
    <xs:enumeration value="bam"/>
    <xs:enumeration value="bgn"/>
    <xs:enumeration value="byr"/>
    <xs:enumeration value="chf"/>
    <xs:enumeration value="czk"/>
    <xs:enumeration value="dkk"/>
    <xs:enumeration value="gbp"/>
  </xs:restriction>
</xs:simpleType>

```

```

<xs:enumeration value="gel"/>
<xs:enumeration value="hrk"/>
<xs:enumeration value="huf"/>
<xs:enumeration value="isk"/>
<xs:enumeration value="ltl"/>
<xs:enumeration value="mdl"/>
<xs:enumeration value="mkd"/>
<xs:enumeration value="nok"/>
<xs:enumeration value="pln"/>
<xs:enumeration value="ron"/>
<xs:enumeration value="rsd"/>
<xs:enumeration value="rub"/>
<xs:enumeration value="sek"/>
<xs:enumeration value="try"/>
<xs:enumeration value="uah"/>
<xs:enumeration value="usd"/>
<xs:enumeration value="other"/>
</xs:restriction>
</xs:simpleType>

```

[top](#)

Simple Type: **DangerousGoodsRegulationsEnum**

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

Sub-types: None

Name DangerousGoodsRegulationsEnum

Content

- Base XSD Type: string
- *value* comes from list:
{'adr'|'iataIcao'|'imoImdg'|'railroadDangerousGoodsBook'}

Documentation Types of dangerous goods regulations.

Schema Component Representation

```

<xs:simpleType name="DangerousGoodsRegulationsEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="adr"/>
    <xs:enumeration value="iataIcao"/>
    <xs:enumeration value="imoImdg"/>
    <xs:enumeration value="railroadDangerousGoodsBook"/>
  </xs:restriction>
</xs:simpleType>

```

[top](#)

Simple Type: **Date**

Super-types: [xs:date](#) < **Date** (by restriction)

Sub-types: None

Name Date

Content

- Base XSD Type: date

Documentation

A combination of year, month and day integer-valued properties plus an optional timezone property. It represents an interval of exactly one day, beginning on the first moment of the day in the timezone, i.e. '00:00:00' up to but not including '24:00:00'.

Schema Component Representation

```
<xs:simpleType name="Date">
  <xs:restriction base="xs:date"/>
</xs:simpleType>
```

[top](#)

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: **DayEnum**

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

Sub-types: None

Name

DayEnum

Content

- Base XSD Type: string
- *value* comes from list:
{'monday'|'tuesday'|'wednesday'|'thursday'|'friday'|'saturday'|'sunday'}

Documentation

Days of the week.

Schema Component Representation

```
<xs:simpleType name="DayEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="monday"/>
    <xs:enumeration value="tuesday"/>
    <xs:enumeration value="wednesday"/>
  </xs:restriction>
</xs:simpleType>
```

```

<xs:enumeration value="thursday"/>
<xs:enumeration value="friday"/>
<xs:enumeration value="saturday"/>
<xs:enumeration value="sunday"/>
</xs:restriction>
</xs:simpleType>

```

[top](#)

Simple Type: **Decimal**

Super-types: [xs:decimal](#) < **Decimal** (by restriction)

Sub-types:

- [AmountOfMoney](#) (by restriction)

Name Decimal

Content

- Base XSD Type: decimal

Documentation A decimal number whose value space is the set of numbers that can be obtained by multiplying an integer by a non-positive power of ten, i.e., expressible as $i \times 10^{-n}$ where i and n are integers and $n \geq 0$.

Schema Component Representation

```

<xs:simpleType name="Decimal">
  <xs:restriction base="xs:decimal"/>
</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'|'n'

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:restriction>
</xs:simpleType>

```

```

<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: **EquipmentTypeEnum**

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

Sub-types: None

Name EquipmentTypeEnum

Content

- Base XSD Type: string
- *value* comes from list:
{'toilet'|'shower'|'informationPoint'|'informatonStele'|'internetTerminal'|'internetWireless'|'payDesk'|'paymentMachine'|'cashMachine'|'vendingMachine'|'faxMachineOrService'|'copyMachineOrService'|'safeDeposit'|'luggageLocker'|'publicPhone'|'publicCoinPhone'|'publicCardPhone'|'elevator'|'picnicFacilities'|'dumpingStation'|'freshWater'|'wasteDisposal'|'refuseBin'|'iceFreeScaffold'|'playground'|'electricChargingStation'|'bikeParking'}

Documentation Equipment available on the parking or parking space or grouped parking spaces.

Schema Component Representation

```

<xs:simpleType name="EquipmentTypeEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="toilet"/>
    <xs:enumeration value="shower"/>
    <xs:enumeration value="informationPoint"/>
    <xs:enumeration value="informatonStele"/>
    <xs:enumeration value="internetTerminal"/>
    <xs:enumeration value="internetWireless"/>
    <xs:enumeration value="payDesk"/>
    <xs:enumeration value="paymentMachine"/>
    <xs:enumeration value="cashMachine"/>
    <xs:enumeration value="vendingMachine"/>
    <xs:enumeration value="faxMachineOrService"/>
    <xs:enumeration value="copyMachineOrService"/>
    <xs:enumeration value="safeDeposit"/>
    <xs:enumeration value="luggageLocker"/>
    <xs:enumeration value="publicPhone"/>
    <xs:enumeration value="publicCoinPhone"/>
    <xs:enumeration value="publicCardPhone"/>
    <xs:enumeration value="elevator"/>
    <xs:enumeration value="picnicFacilities"/>
    <xs:enumeration value="dumpingStation"/>
    <xs:enumeration value="freshWater"/>
    <xs:enumeration value="wasteDisposal"/>
    <xs:enumeration value="refuseBin"/>
    <xs:enumeration value="iceFreeScaffold"/>
    <xs:enumeration value="playground"/>
    <xs:enumeration value="electricChargingStation"/>
    <xs:enumeration value="bikeParking"/>
  </xs:restriction>
</xs:simpleType>

```

```

<xs:enumeration value="tollTerminal"/>
<xs:enumeration value="defibrillator"/>
<xs:enumeration value="firstAidEquipment"/>
<xs:enumeration value="fireHose"/>
<xs:enumeration value="fireExtinguisher"/>
<xs:enumeration value="fireHydrant"/>
<xs:enumeration value="none"/>
<xs:enumeration value="unknown"/>
<xs:enumeration value="other"/>
</xs:restriction>
</xs:simpleType>

```

[top](#)

Simple Type: **Float**

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

Sub-types:

- [Ampere](#) (by restriction)
- [CubicMetres](#) (by restriction)
- [MetresAsFloat](#) (by restriction)
- [Percentage](#) (by restriction)
- [Seconds](#) (by restriction)
- [TemperatureCelsius](#) (by restriction)
- [Tonnes](#) (by restriction)
- [Volt](#) (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: **FuelTypeEnum**

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

Sub-types: None

Name FuelTypeEnum

Content

- Base XSD Type: string
- *value* comes from list:
{'battery'|'biodiesel'|'diesel'|'dieselBatteryHybrid'|'ethanol'|'hydrogen'|'liquidGa

Documentation Type of fuel used by a vehicle.

Schema Component Representation

```
<xs:simpleType name="FuelTypeEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="battery"/>
    <xs:enumeration value="biodiesel"/>
    <xs:enumeration value="diesel"/>
    <xs:enumeration value="dieselBatteryHybrid"/>
    <xs:enumeration value="ethanol"/>
    <xs:enumeration value="hydrogen"/>
    <xs:enumeration value="liquidGas"/>
    <xs:enumeration value="lpg"/>
    <xs:enumeration value="methane"/>
    <xs:enumeration value="petrol"/>
    <xs:enumeration value="petrolBatteryHybrid"/>
  </xs:restriction>
</xs:simpleType>
```

[top](#)

Simple Type: **GroupOfParkingSitesTypeEnum**

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

Sub-types: None

Name GroupOfParkingSitesTypeEnum

Content

- Base XSD Type: string
- *value* comes from list:
{'truckParkingPriorityZone'|'aggregationOfInformation'}

Documentation The type of this group of parking sites.

Schema Component Representation

```
<xs:simpleType name="GroupOfParkingSitesTypeEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="truckParkingPriorityZone"/>
    <xs:enumeration value="aggregationOfInformation"/>
  </xs:restriction>
</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: **IndexReference**

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

Sub-types: None

Name IndexReference

Content

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

Documentation A reference to an object given by its index qualifier.

Schema Component Representation

```
<xs:simpleType name="IndexReference">
  <xs:restriction base="D2LogicalModel:String"/>
</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:enumeration value="test"/>
  </xs:restriction>
</xs:simpleType>
```

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>
```

Simple Type: **InterUrbanParkingSiteLocationEnum**

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

Sub-types: None

Name InterUrbanParkingSiteLocationEnum

Content

- Base XSD Type: string
- *value* comes from list:
{'motorway'|'nearbyMotorway'|'layBy'|'onStreet'|'other'}

Documentation Location of the truck or motorway related parking.

Schema Component Representation

```
<xs:simpleType name="InterUrbanParkingSiteLocationEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="motorway"/>
    <xs:enumeration value="nearbyMotorway"/>
    <xs:enumeration value="layBy"/>
    <xs:enumeration value="onStreet"/>
    <xs:enumeration value="other"/>
  </xs:restriction>
</xs:simpleType>
```

Simple Type: **JunctionClassificationEnum**

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

Sub-types: None

Name	JunctionClassificationEnum
Content	• Base XSD Type: string • <i>value</i> comes from list: {'threeWayInterchange' 'interchange' 'motorwayConnection' 'junction' 'tempor
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: LABELSecurityLevelEnum

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

Name	LABELSecurityLevelEnum
Content	• Base XSD Type: string • <i>value</i> comes from list: {'none' 'securityLevel1' 'securityLevel2' 'securityLevel3' 'securityLevel4' 'secu
Documentation	Security level defined by the LABEL project http://truckparkinglabel.eu .

Schema Component Representation

```
<xs:simpleType name="LABELSecurityLevelEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="none"/>
    <xs:enumeration value="securityLevel1"/>
    <xs:enumeration value="securityLevel2"/>
    <xs:enumeration value="securityLevel3"/>
    <xs:enumeration value="securityLevel4"/>
    <xs:enumeration value="securityLevel5"/>
    <xs:enumeration value="unknown"/>
  </xs:restriction>
</xs:simpleType>
```

[top](#)

Simple Type: LABELServiceLevelEnum

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

Sub-types: None

Name LABELServiceLevelEnum

Content

- Base XSD Type: string
- *value* comes from list:
{'none'|'serviceLevel1'|'serviceLevel2'|'serviceLevel3'|'serviceLevel4'|'serviceLevel5'|'unknown'}

Documentation Service level defined by the LABEL project <http://truckparkinglabel.eu>.

Schema Component Representation

```
<xs:simpleType name="LABELServiceLevelEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="none"/>
    <xs:enumeration value="serviceLevel1"/>
    <xs:enumeration value="serviceLevel2"/>
    <xs:enumeration value="serviceLevel3"/>
    <xs:enumeration value="serviceLevel4"/>
    <xs:enumeration value="serviceLevel5"/>
    <xs:enumeration value="unknown"/>
  </xs:restriction>
</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:
{'allLanesCompleteCarriageway'|'busLane'|'busStop'|'carPoolLane'|'centralReservation'|'crawlerLane'|'emergencyLane'|'escapeLane'|'expressLane'|'hardShoulder'|'heavyVehicleLane'|'lane1'|'lane2'}

Documentation List of descriptors identifying specific lanes.

Schema Component Representation

```
<xs:simpleType name="LaneEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="allLanesCompleteCarriageway"/>
    <xs:enumeration value="busLane"/>
    <xs:enumeration value="busStop"/>
    <xs:enumeration value="carPoolLane"/>
    <xs:enumeration value="centralReservation"/>
    <xs:enumeration value="crawlerLane"/>
    <xs:enumeration value="emergencyLane"/>
    <xs:enumeration value="escapeLane"/>
    <xs:enumeration value="expressLane"/>
    <xs:enumeration value="hardShoulder"/>
    <xs:enumeration value="heavyVehicleLane"/>
    <xs:enumeration value="lane1"/>
    <xs:enumeration value="lane2"/>
  </xs:restriction>
</xs:simpleType>
```

```

<xs:enumeration value="lane3"/>
<xs:enumeration value="lane4"/>
<xs:enumeration value="lane5"/>
<xs:enumeration value="lane6"/>
<xs:enumeration value="lane7"/>
<xs:enumeration value="lane8"/>
<xs:enumeration value="lane9"/>
<xs:enumeration value="layBy"/>
<xs:enumeration value="leftHandTurningLane"/>
<xs:enumeration value="leftLane"/>
<xs:enumeration value="localTrafficLane"/>
<xs:enumeration value="middleLane"/>
<xs:enumeration value="opposingLanes"/>
<xs:enumeration value="overtakingLane"/>
<xs:enumeration value="rightHandTurningLane"/>
<xs:enumeration value="rightLane"/>
<xs:enumeration value="rushHourLane"/>
<xs:enumeration value="setDownArea"/>
<xs:enumeration value="slowVehicleLane"/>
<xs:enumeration value="throughTrafficLane"/>
<xs:enumeration value="tidalFlowLane"/>
<xs:enumeration value="turningLane"/>
<xs:enumeration value="verge"/>
</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: LoadType2Enum

Super-types: [xs:string](#) < **LoadType2Enum** (by restriction)

Sub-types: None

Name

LoadType2Enum

Content

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

Documentation

Loads that are currently not supported in loadType.

Schema Component Representation

```
<xs:simpleType name="LoadType2Enum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="refrigeratedGoods"/>
  </xs:restriction>
</xs:simpleType>
```

[top](#)

Simple Type: LoadTypeEnum

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

Sub-types: None

Name

LoadTypeEnum

Content

- Base XSD Type: string
- *value* comes from list:
{'abnormalLoad'|'ammunition'|'chemicals'|'combustibleMaterials'|'corrosiveMa'

Documentation

Types of load carried by a vehicle.

Schema Component Representation

```
<xs:simpleType name="LoadTypeEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="abnormalLoad"/>
    <xs:enumeration value="ammunition"/>
    <xs:enumeration value="chemicals"/>
    <xs:enumeration value="combustibleMaterials"/>
    <xs:enumeration value="corrosiveMaterials"/>
    <xs:enumeration value="debris"/>
    <xs:enumeration value="empty"/>
    <xs:enumeration value="explosiveMaterials"/>
    <xs:enumeration value="extraHighLoad"/>
    <xs:enumeration value="extraLongLoad"/>
    <xs:enumeration value="extraWideLoad"/>
    <xs:enumeration value="fuel"/>
    <xs:enumeration value="glass"/>
    <xs:enumeration value="goods"/>
    <xs:enumeration value="hazardousMaterials"/>
    <xs:enumeration value="liquid"/>
    <xs:enumeration value="livestock"/>
    <xs:enumeration value="materials"/>
    <xs:enumeration value="materialsDangerousForPeople"/>
    <xs:enumeration value="materialsDangerousForTheEnvironment"/>
    <xs:enumeration value="materialsDangerousForWater"/>
    <xs:enumeration value="oil"/>
    <xs:enumeration value="ordinary"/>
    <xs:enumeration value="perishableProducts"/>
    <xs:enumeration value="petrol"/>
    <xs:enumeration value="pharmaceuticalMaterials"/>
  </xs:restriction>
</xs:simpleType>
```

```

<xs:enumeration value="radioactiveMaterials"/>
<xs:enumeration value="refuse"/>
<xs:enumeration value="toxicMaterials"/>
<xs:enumeration value="vehicles"/>
<xs:enumeration value="other"/>
</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'

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: **MeansOfPaymentEnum**

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

Sub-types: None

Name MeansOfPaymentEnum

Content

- Base XSD Type: string
- *value* comes from list:
{'paymentCard'|'cash'|'cashCoinsOnly'|'directCashTransfer'|'electronicSettlen

Documentation Means of payment

Schema Component Representation

```
<xs:simpleType name="MeansOfPaymentEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="paymentCard"/>
    <xs:enumeration value="cash"/>
    <xs:enumeration value="cashCoinsOnly"/>
    <xs:enumeration value="directCashTransfer"/>
    <xs:enumeration value="electronicSettlement"/>
    <xs:enumeration value="rfid"/>
    <xs:enumeration value="mobileApp"/>
    <xs:enumeration value="payBySMS"/>
    <xs:enumeration value="mobilePhone"/>
    <xs:enumeration value="unknown"/>
    <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

Name MetresAsNonNegativeInteger

Content

- Base XSD Type: nonNegativeInteger

Documentation 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: **MonthOfYearEnum**

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

Sub-types: None

Name MonthOfYearEnum

Content

- Base XSD Type: string
- *value* comes from list:
{'january'|'february'|'march'|'april'|'may'|'june'|'july'|'august'|'september'|'october'}

Documentation A list of the months of the year.

Schema Component Representation

```
<xs:simpleType name="MonthOfYearEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="january"/>
    <xs:enumeration value="february"/>
    <xs:enumeration value="march"/>
    <xs:enumeration value="april"/>
    <xs:enumeration value="may"/>
    <xs:enumeration value="june"/>
    <xs:enumeration value="july"/>
    <xs:enumeration value="august"/>
    <xs:enumeration value="september"/>
    <xs:enumeration value="october"/>
    <xs:enumeration value="november"/>
    <xs:enumeration value="december"/>
  </xs:restriction>
</xs:simpleType>
```

[top](#)

Simple Type: **MultilingualStringValue**

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

Sub-types:

- [MultilingualStringValue](#) (by extension)

Name MultilingualStringValue

Content

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

Schema Component Representation

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

[top](#)

Simple Type: **NonNegativeInteger**

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

Sub-types:

- [AlertCLocationCode](#) (by restriction)
- [MetresAsNonNegativeInteger](#) (by restriction)
- [SquareMetres](#) (by restriction)

Name NonNegativeInteger

Content

- Base XSD Type: nonNegativeInteger

Documentation 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: **OccupancyDetectionTypeEnum**

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

Sub-types: None

Name OccupancyDetectionTypeEnum

Content

- Base XSD Type: string
- *value* comes from list:
{'none'|'balancing'|'singleSpaceDetection'|'modelBased'|'manual'|'unspecified'}

Documentation Type of parking occupancy detection (balancing, single slot, ...).

Schema Component Representation

```
<xs:simpleType name="OccupancyDetectionTypeEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="none"/>
    <xs:enumeration value="balancing"/>
    <xs:enumeration value="singleSpaceDetection"/>
    <xs:enumeration value="modelBased"/>
    <xs:enumeration value="manual"/>
    <xs:enumeration value="unspecified"/>
    <xs:enumeration value="other"/>
    <xs:enumeration value="unknown"/>
  </xs:restriction>
</xs:simpleType>
```

[top](#)

Simple Type: **OwnershipTypeEnum**

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

Sub-types: None

Name OwnershipTypeEnum

Content

- Base XSD Type: string
- *value* comes from list:
{'public'|'private'|'publicPrivate'|'resident'|'unknown'|'other'}

Documentation Ownership type enum.

Schema Component Representation

```
<xs:simpleType name="OwnershipTypeEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="public"/>
    <xs:enumeration value="private"/>
    <xs:enumeration value="publicPrivate"/>
    <xs:enumeration value="resident"/>
    <xs:enumeration value="unknown"/>
    <xs:enumeration value="other"/>
  </xs:restriction>
</xs:simpleType>
```

[top](#)

Simple Type: **ParkingDurationEnum**

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

Sub-types: None

Name ParkingDurationEnum

Content

- Base XSD Type: string
- *value* comes from list:
{'pickUpDropOff'|'shortTerm'|'shortTerm24hours'|'shortTerm48hours'|'shortTer

Documentation Parking durations.

Schema Component Representation

```
<xs:simpleType name="ParkingDurationEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="pickUpDropOff"/>
    <xs:enumeration value="shortTerm"/>
    <xs:enumeration value="shortTerm24hours"/>
    <xs:enumeration value="shortTerm48hours"/>
    <xs:enumeration value="shortTerm72hours"/>
    <xs:enumeration value="shortTerm96hours"/>
    <xs:enumeration value="longTerm"/>
    <xs:enumeration value="unknown"/>
    <xs:enumeration value="other"/>
  </xs:restriction>
</xs:simpleType>
```

[top](#)

Simple Type: **ParkingLayoutEnum**

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

Sub-types: None

Name ParkingLayoutEnum

Content

- Base XSD Type: string
- *value* comes from list:
{'multiStorey'|'singleLevel'|'underground'|'undergroundAndMultistorey'|'autom

Documentation Types of layout of the parking site.

Schema Component Representation

```
<xs:simpleType name="ParkingLayoutEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="multiStorey"/>
    <xs:enumeration value="singleLevel"/>
    <xs:enumeration value="underground"/>
    <xs:enumeration value="undergroundAndMultistorey"/>
    <xs:enumeration value="automatedParkingGarage"/>
    <xs:enumeration value="openSpace"/>
    <xs:enumeration value="covered"/>
    <xs:enumeration value="nested"/>
    <xs:enumeration value="field"/>
    <xs:enumeration value="unknown"/>
    <xs:enumeration value="other"/>
  </xs:restriction>
</xs:simpleType>
```

[top](#)

Simple Type: **ParkingModeEnum**

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

Sub-types: None

Name ParkingModeEnum

Content

- Base XSD Type: string
- *value* comes from list:
{'perpendicularParking'|'parallelParking'|'echelonParking'|'parkingOnOppositeSideOfRoad'|'other'}

Documentation The arrangement of the parking space or the group of parking spaces in relation to the

Schema Component Representation

```
<xs:simpleType name="ParkingModeEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="perpendicularParking"/>
    <xs:enumeration value="parallelParking"/>
    <xs:enumeration value="echelonParking"/>
    <xs:enumeration value="parkingOnOppositeSideOfRoad"/>
    <xs:enumeration value="other"/>
  </xs:restriction>
</xs:simpleType>
```

[top](#)

Simple Type: **ParkingPaymentModeEnum**

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

Sub-types: None

Name ParkingPaymentModeEnum

Content

- Base XSD Type: string
- *value* comes from list:
{'payAndDisplay'|'payManualAtExitBooth'|'payPriorToExit'|'payByPrepaidToken'|'payAndExit'|'other'}

Documentation Mode of payment for parking.

Schema Component Representation

```
<xs:simpleType name="ParkingPaymentModeEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="payAndDisplay"/>
    <xs:enumeration value="payManualAtExitBooth"/>
    <xs:enumeration value="payPriorToExit"/>
    <xs:enumeration value="payByPrepaidToken"/>
    <xs:enumeration value="payAndExit"/>
    <xs:enumeration value="other"/>
  </xs:restriction>
</xs:simpleType>
```

[top](#)

Simple Type: **ParkingRouteDirectionEnum**

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

Sub-types: None

Name ParkingRouteDirectionEnum

Content

- Base XSD Type: string
- *value* comes from list:
{'towardsParkingSite'|'awayFromParkingSite'}

Documentation The direction of the parking route.

Schema Component Representation

```
<xs:simpleType name="ParkingRouteDirectionEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="towardsParkingSite"/>
    <xs:enumeration value="awayFromParkingSite"/>
  </xs:restriction>
</xs:simpleType>
```

[top](#)

Simple Type: **ParkingRouteTypeEnum**

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

Sub-types: None

Name ParkingRouteTypeEnum

Content

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

Documentation The type of the parking route.

Schema Component Representation

```
<xs:simpleType name="ParkingRouteTypeEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="lorry"/>
    <xs:enumeration value="other"/>
  </xs:restriction>
</xs:simpleType>
```

[top](#)

Simple Type: **ParkingSecurityEnum**

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

Sub-types: None

Name ParkingSecurityEnum

Content

- Base XSD Type: string
- *value* comes from list:
{'socialControl'|'securityStaff'|'externalSecurity'|'cctv'|'dog'|'guard24hours'|'lighting'|'floodLight'|'fences'|'areaSeparatedFromSurroundings'|'none'|'unknown'|'other'}

Documentation Specifies security measures related to the parking site or particular spaces.

Schema Component Representation

```
<xs:simpleType name="ParkingSecurityEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="socialControl"/>
    <xs:enumeration value="securityStaff"/>
    <xs:enumeration value="externalSecurity"/>
    <xs:enumeration value="cctv"/>
    <xs:enumeration value="dog"/>
    <xs:enumeration value="guard24hours"/>
    <xs:enumeration value="lighting"/>
    <xs:enumeration value="floodLight"/>
    <xs:enumeration value="fences"/>
    <xs:enumeration value="areaSeparatedFromSurroundings"/>
    <xs:enumeration value="none"/>
    <xs:enumeration value="unknown"/>
    <xs:enumeration value="other"/>
  </xs:restriction>
</xs:simpleType>
```

[top](#)

Simple Type: **ParkingSiteStatusEnum**

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

Sub-types: None

Name ParkingSiteStatusEnum

Content

- Base XSD Type: string
- *value* comes from list:
{'spacesAvailable'|'almostFull'|'fullAtEntrance'|'full'|'unknown'|'other'}

Documentation The status of the parking site (spaces available or not).

Schema Component Representation

```
<xs:simpleType name="ParkingSiteStatusEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="spacesAvailable"/>
    <xs:enumeration value="almostFull"/>
    <xs:enumeration value="fullAtEntrance"/>
    <xs:enumeration value="full"/>
    <xs:enumeration value="unknown"/>
    <xs:enumeration value="other"/>
  </xs:restriction>
</xs:simpleType>
```

[top](#)

Simple Type: **ParkingSpaceAccessibilityEnum**

<i>Super-types:</i>	xs:string < ParkingSpaceAccessibilityEnum (by restriction)
<i>Sub-types:</i>	None

Name ParkingSpaceAccessibilityEnum

Content

- Base XSD Type: string
- *value* comes from list: {'extraSpaceLeftSide'|'extraSpaceRightSide'|'nearbyPedestrianExit'|'bordersMarked'|'other'}

Documentation Easements for handicapped people especially related to a parking space or a group of spaces.

Schema Component Representation

```
<xs:simpleType name="ParkingSpaceAccessibilityEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="extraSpaceLeftSide"/>
    <xs:enumeration value="extraSpaceRightSide"/>
    <xs:enumeration value="nearbyPedestrianExit"/>
    <xs:enumeration value="bordersMarked"/>
    <xs:enumeration value="other"/>
  </xs:restriction>
</xs:simpleType>
```

[top](#)

Simple Type: **ParkingSpacePhysicsEnum**

<i>Super-types:</i>	xs:string < ParkingSpacePhysicsEnum (by restriction)
<i>Sub-types:</i>	None

Name ParkingSpacePhysicsEnum

Content

- Base XSD Type: string
- *value* comes from list: {'driveThrough'|'openAir'}

Documentation Specifies drive through and open air properties for the parking space or the group of parking spaces.

Schema Component Representation

```
<xs:simpleType name="ParkingSpacePhysicsEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="driveThrough"/>
    <xs:enumeration value="openAir"/>
  </xs:restriction>
</xs:simpleType>
```

[top](#)

Simple Type: **ParkingSpecialLocationEnum**

<i>Super-types:</i>	xs:string < ParkingSpecialLocationEnum (by restriction)
<i>Sub-types:</i>	None

Name	ParkingSpecialLocationEnum
Content	• Base XSD Type: string • <i>value</i> comes from list: {'airportTerminal' 'exhibitonCentre' 'shoppingCentre' 'specificFacility' 'trainSta
Documentation	Locations, often associated with a building, for a SpecialLocationParkingSite.

Schema Component Representation

```
<xs:simpleType name="ParkingSpecialLocationEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="airportTerminal"/>
    <xs:enumeration value="exhibitonCentre"/>
    <xs:enumeration value="shoppingCentre"/>
    <xs:enumeration value="specificFacility"/>
    <xs:enumeration value="trainStation"/>
    <xs:enumeration value="campground"/>
    <xs:enumeration value="themePark"/>
    <xs:enumeration value="ferryTerminal"/>
    <xs:enumeration value="vehicleOnRailTerminal"/>
    <xs:enumeration value="coachStation"/>
    <xs:enumeration value="cableCarStation"/>
    <xs:enumeration value="publicTransportStation"/>
    <xs:enumeration value="market"/>
    <xs:enumeration value="religiousCentre"/>
    <xs:enumeration value="conventionCentre"/>
    <xs:enumeration value="cinema"/>
    <xs:enumeration value="skilift"/>
    <xs:enumeration value="unknown"/>
    <xs:enumeration value="other"/>
  </xs:restriction>
</xs:simpleType>
```

[top](#)

Simple Type: **ParkingSupervisionEnum**

<i>Super-types:</i>	xs:string < ParkingSupervisionEnum (by restriction)
<i>Sub-types:</i>	None

Name	ParkingSupervisionEnum
Content	• Base XSD Type: string • <i>value</i> comes from list: {'remote' 'onSite' 'controlCentreOnSite' 'controlCentreOffSite' 'patrol' 'none' 'u
Documentation	Defines the kind of supervision of the parking site.

Schema Component Representation

```
<xs:simpleType name="ParkingSupervisionEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="remote"/>
    <xs:enumeration value="onSite"/>
    <xs:enumeration value="controlCentreOnSite"/>
    <xs:enumeration value="controlCentreOffSite"/>
    <xs:enumeration value="patrol"/>
    <xs:enumeration value="none"/>
  </xs:restriction>
</xs:simpleType>
```

```

        <xs:enumeration value="unknown"/>
        <xs:enumeration value="other"/>
    </xs:restriction>
</xs:simpleType>

```

[top](#)

Simple Type: **ParkingTypeOfGroup**

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

Sub-types: None

Name ParkingTypeOfGroup

Content

- Base XSD Type: string
- *value* comes from list:
{'adjacentSpaces'|'nonAdjacentSpaces'|'completeFloor'|'mixedUsage'|'statisticsOnly'|'singleParameters'|'other'}

Documentation The type of group specification (group of parking spaces).

Schema Component Representation

```

<xs:simpleType name="ParkingTypeOfGroup">
  <xs:restriction base="xs:string">
    <xs:enumeration value="adjacentSpaces"/>
    <xs:enumeration value="nonAdjacentSpaces"/>
    <xs:enumeration value="completeFloor"/>
    <xs:enumeration value="mixedUsage"/>
    <xs:enumeration value="statisticsOnly"/>
    <xs:enumeration value="singleParameters"/>
    <xs:enumeration value="other"/>
  </xs:restriction>
</xs:simpleType>

```

[top](#)

Simple Type: **ParkingUsageScenarioEnum**

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

Sub-types: None

Name ParkingUsageScenarioEnum

Content

- Base XSD Type: string
- *value* comes from list:
{'truckParking'|'restArea'|'serviceArea'|'loadingBay'|'overnightParking'|'unknown'}

Documentation Types of parking usage (park & ride, kiss & ride, ...)

Schema Component Representation

```

<xs:simpleType name="ParkingUsageScenarioEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="truckParking"/>
    <xs:enumeration value="restArea"/>
    <xs:enumeration value="serviceArea"/>
  </xs:restriction>
</xs:simpleType>

```

```

<xs:enumeration value="loadingBay"/>
<xs:enumeration value="overnightParking"/>
<xs:enumeration value="unknown"/>
<xs:enumeration value="other"/>
</xs:restriction>
</xs:simpleType>

```

[top](#)

Simple Type: **PaymentCardBrandsEnum**

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

Name PaymentCardBrandsEnum

Content

- Base XSD Type: string
- *value* comes from list:
{americanExpress|cirrus|dinersClub|discoverCard|giroCard|maestro|m:

Documentation Brands of payment cards.

Schema Component Representation

```

<xs:simpleType name="PaymentCardBrandsEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="americanExpress"/>
    <xs:enumeration value="cirrus"/>
    <xs:enumeration value="dinersClub"/>
    <xs:enumeration value="discoverCard"/>
    <xs:enumeration value="giroCard"/>
    <xs:enumeration value="maestro"/>
    <xs:enumeration value="masterCard"/>
    <xs:enumeration value="visa"/>
    <xs:enumeration value="vPay"/>
    <xs:enumeration value="other"/>
  </xs:restriction>
</xs:simpleType>

```

[top](#)

Simple Type: **PaymentCardTypesEnum**

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

Name PaymentCardTypesEnum

Content

- Base XSD Type: string
- *value* comes from list:
{creditCard|debitCard|chargeCard|fleetCard|storedValueCard|other}

Documentation Types of payment cards.

Schema Component Representation

```

<xs:simpleType name="PaymentCardTypesEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="creditCard"/>
    <xs:enumeration value="debitCard"/>
    <xs:enumeration value="chargeCard"/>
    <xs:enumeration value="fleetCard"/>
    <xs:enumeration value="storedValueCard"/>
    <xs:enumeration value="other"/>
  </xs:restriction>
</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: **PermitTypeEnum**

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

Sub-types: None

Name PermitTypeEnum

Content

- Base XSD Type: string
- *value* comes from list:
{'blueZonePermit'|'careTakingPermit'|'carpoolingPermit'|'carSharingPermit'|'d

Documentation Type of permission for parking.

Schema Component Representation

```

<xs:simpleType name="PermitTypeEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="blueZonePermit"/>
    <xs:enumeration value="careTakingPermit"/>
    <xs:enumeration value="carpoolingPermit"/>
    <xs:enumeration value="carSharingPermit"/>
    <xs:enumeration value="disabledPermit"/>
    <xs:enumeration value="emergencyVehiclePermit"/>
    <xs:enumeration value="employeePermit"/>
  </xs:restriction>
</xs:simpleType>

```

```

<xs:enumeration value="fairPermit"/>
<xs:enumeration value="governmentPermit"/>
<xs:enumeration value="maintenanceVehiclePermit"/>
<xs:enumeration value="residentPermit"/>
<xs:enumeration value="roadWorksPermit"/>
<xs:enumeration value="specificIdentifiedVehiclePermit"/>
<xs:enumeration value="taxiPermit"/>
<xs:enumeration value="other"/>
</xs:restriction>
</xs:simpleType>

```

[top](#)

Simple Type: **PublicHolidayTypeEnum**

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

Name PublicHolidayTypeEnum

Content

- Base XSD Type: string
- *value* comes from list:
{'betweenChristmasAndNewYear'|'boxingDay'|'bridgeHoliday'|'christmasEve'}

Documentation Types of public holiday.

Schema Component Representation

```

<xs:simpleType name="PublicHolidayTypeEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="betweenChristmasAndNewYear"/>
    <xs:enumeration value="boxingDay"/>
    <xs:enumeration value="bridgeHoliday"/>
    <xs:enumeration value="christmasEve"/>
    <xs:enumeration value="christmasDayAndBoxingDay"/>
    <xs:enumeration value="christmasHolidayPeriod"/>
    <xs:enumeration value="dayFollowingPublicHoliday"/>
    <xs:enumeration value="easterFridayHoliday"/>
    <xs:enumeration value="easterHolidayPeriod"/>
    <xs:enumeration value="easterMondayHoliday"/>
    <xs:enumeration value="easterSaturday"/>
    <xs:enumeration value="easterSunday"/>
    <xs:enumeration value="eveOfPublicHoliday"/>
    <xs:enumeration value="holidayPeriod"/>
    <xs:enumeration value="inLieuOfPublicHoliday"/>
    <xs:enumeration value="january2ndHoliday"/>
    <xs:enumeration value="newYearsDay"/>
    <xs:enumeration value="newYearsEve"/>
    <xs:enumeration value="notPublicHoliday"/>
    <xs:enumeration value="publicHoliday"/>
    <xs:enumeration value="other"/>
  </xs:restriction>
</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: **RegulationEnum**

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

Sub-types: None

Name RegulationEnum

Content

- Base XSD Type: string
- *value* comes from list:
{'permitted'|'prohibited'|'punishable'|'seasonalHeterogeneous'|'permittedOnly'.

Documentation Regulation parameters for actions.

Schema Component Representation

```
<xs:simpleType name="RegulationEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="permitted"/>
    <xs:enumeration value="prohibited"/>
    <xs:enumeration value="punishable"/>
    <xs:enumeration value="seasonalHeterogeneous"/>
    <xs:enumeration value="permittedOnlyAtParticularTimes"/>
    <xs:enumeration value="permittedOnlyOnParticularAreas"/>
    <xs:enumeration value="prohibitedAtParticularTimes"/>
    <xs:enumeration value="prohibitedOnParticularAreas"/>
    <xs:enumeration value="onlyOnRequest"/>
    <xs:enumeration value="heterogeneous"/>
    <xs:enumeration value="onlyOutsideBuildings"/>
    <xs:enumeration value="onlyInsideBuildings"/>
    <xs:enumeration value="unspecified"/>
    <xs:enumeration value="unknown"/>
    <xs:enumeration value="other"/>
  </xs:restriction>
</xs:simpleType>
```

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

[top](#)

Simple Type: **ReservationTypeEnum**

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

Sub-types: None

Name ReservationTypeEnum

Content

- Base XSD Type: string
- *value* comes from list:
{'optional'|'mandatory'|'notAvailable'|'partly'|'unknown'|'unspecified'}

Documentation Reservation type enum.

Schema Component Representation

```
<xs:simpleType name="ReservationTypeEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="optional"/>
    <xs:enumeration value="mandatory"/>
    <xs:enumeration value="notAvailable"/>
    <xs:enumeration value="partly"/>
    <xs:enumeration value="unknown"/>
    <xs:enumeration value="unspecified"/>
  </xs:restriction>
</xs:simpleType>
```

[top](#)

Simple Type: **ResponseEnum**

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

Sub-types: None

Name ResponseEnum

Content

- Base XSD Type: string
- *value* comes from list:
{'acknowledge'|'requestDenied'|'subscriptionRequestDenied'}

Documentation Types of response that a supplier can return to a requesting client.

Schema Component Representation

```
<xs:simpleType name="ResponseEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="acknowledge"/>
    <xs:enumeration value="requestDenied"/>
    <xs:enumeration value="subscriptionRequestDenied"/>
  </xs:restriction>
</xs:simpleType>
```

Simple Type: **RestAreaActivityEnum**

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

Sub-types: None

Name RestAreaActivityEnum

Content

- Base XSD Type: string
- *value* comes from list:
{'openFire'|'overnightParking'|'picnic'|'smoking'|'camping'|'handlingHazardous'}

Documentation Rest area activity enum.

Schema Component Representation

```
<xs:simpleType name="RestAreaActivityEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="openFire"/>
    <xs:enumeration value="overnightParking"/>
    <xs:enumeration value="picnic"/>
    <xs:enumeration value="smoking"/>
    <xs:enumeration value="camping"/>
    <xs:enumeration value="handlingHazardousMaterial"/>
    <xs:enumeration value="barbecue"/>
    <xs:enumeration value="other"/>
  </xs:restriction>
</xs:simpleType>
```

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>
```

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>
```

Simple Type: **ServiceFacilityTypeEnum**

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

Name ServiceFacilityTypeEnum
Content

- Base XSD Type: string
- *value* comes from list:
 {'hotel'|'motel'|'overnightAccommodation'|'shop'|'kiosk'|'foodShopping'|'cafe'|'r

Documentation Service facilities available on the parking site, parking space or group of parking space

Schema Component Representation

```
<xs:simpleType name="ServiceFacilityTypeEnum">  
  <xs:restriction base="xs:string">  
    <xs:enumeration value="hotel"/>  
    <xs:enumeration value="motel"/>  
    <xs:enumeration value="overnightAccommodation"/>  
    <xs:enumeration value="shop"/>  
    <xs:enumeration value="kiosk"/>  
    <xs:enumeration value="foodShopping"/>  
    <xs:enumeration value="cafe"/>  
    <xs:enumeration value="restaurant"/>  
    <xs:enumeration value="restaurantSelfService"/>  
    <xs:enumeration value="motorwayRestaurant"/>  
    <xs:enumeration value="motorwayRestaurantSmall"/>  
    <xs:enumeration value="sparePartsShopping"/>  
    <xs:enumeration value="petrolStation"/>  
    <xs:enumeration value="vehicleMaintenance"/>  
    <xs:enumeration value="tyreRepair"/>  
    <xs:enumeration value="truckRepair"/>  
    <xs:enumeration value="truckWash"/>  
    <xs:enumeration value="carWash"/>  
    <xs:enumeration value="pharmacy"/>  
  </xs:restriction>  
</xs:simpleType>
```

```

<xs:enumeration value="medicalFacility"/>
<xs:enumeration value="police"/>
<xs:enumeration value="touristInformation"/>
<xs:enumeration value="bikeSharing"/>
<xs:enumeration value="docstop"/>
<xs:enumeration value="laundry"/>
<xs:enumeration value="leisureActivities"/>
<xs:enumeration value="unknown"/>
<xs:enumeration value="other"/>
</xs:restriction>
</xs:simpleType>

```

[top](#)

Simple Type: **SpecialDayTypeEnum**

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

Sub-types: None

Name SpecialDayTypeEnum

Content

- Base XSD Type: string
- *value* comes from list:
{'bicycleRaceDay'|'bullFightDay'|'carnivalDay'|'exhibitionDay'|'festivalDay'|'ga

Documentation Collection of general types of days.

Schema Component Representation

```

<xs:simpleType name="SpecialDayTypeEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="bicycleRaceDay"/>
    <xs:enumeration value="bullFightDay"/>
    <xs:enumeration value="carnivalDay"/>
    <xs:enumeration value="exhibitionDay"/>
    <xs:enumeration value="festivalDay"/>
    <xs:enumeration value="gamesDay"/>
    <xs:enumeration value="horseRaceMeetingDay"/>
    <xs:enumeration value="huntMeetingDay"/>
    <xs:enumeration value="marathonRaceDay"/>
    <xs:enumeration value="marketDay"/>
    <xs:enumeration value="motorSportRaceMeetingDay"/>
    <xs:enumeration value="nonWorkingDay"/>
    <xs:enumeration value="raceMeetingDay"/>
    <xs:enumeration value="regattaDay"/>
    <xs:enumeration value="showDay"/>
    <xs:enumeration value="sportsMeetingDay"/>
    <xs:enumeration value="workingDay"/>
    <xs:enumeration value="schoolDay"/>
    <xs:enumeration value="electionDay"/>
    <xs:enumeration value="publicHoliday"/>
    <xs:enumeration value="holidays"/>
    <xs:enumeration value="undefinedDayType"/>
    <xs:enumeration value="unknown"/>
    <xs:enumeration value="other"/>
  </xs:restriction>
</xs:simpleType>

```

[top](#)

Simple Type: **SquareMetres**

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

Sub-types: None

Name SquareMetres

Content

- Base XSD Type: nonNegativeInteger

Documentation Square metres.

Schema Component Representation

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

[top](#)

Simple Type: **String**

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

Sub-types:

- [IndexReference](#) (by restriction)

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: **TemperatureCelsius**

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

Sub-types: None

Name TemperatureCelsius

Content

- Base XSD Type: float

Documentation

A measure of temperature defined in degrees Celsius.

Schema Component Representation

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

[top](#)

Simple Type: Time

Super-types: [xs:time](#) < **Time** (by restriction)

Sub-types: None

Name Time

Content

- Base XSD Type: time

Documentation

An instant of time that recurs every day. The value space of time is the space of time of day values as defined in § 5.3 of [ISO 8601]. Specifically, it is a set of zero-duration daily time instances.

Schema Component Representation

```
<xs:simpleType name="Time">
  <xs:restriction base="xs:time"/>
</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: TpegLoc01AreaLocationSubtypeEnum

Super-types: [xs:string](#) < **TpegLoc01AreaLocationSubtypeEnum** (by restriction)

Sub-types: None

Name TpegLoc01AreaLocationSubtypeEnum

Content

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

Documentation Types of area.

Schema Component Representation

```
<xs:simpleType name="TpegLoc01AreaLocationSubtypeEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="largeArea"/>
    <xs:enumeration value="other"/>
  </xs:restriction>
</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'}

Schema Component Representation

```
<xs:simpleType name="TpegLoc01LinearLocationSubtypeEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="segment"/>
  </xs:restriction>
</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: TpegLoc03AreaDescriptorSubtypeEnum**

Super-types: [xs:string](#) < **TpegLoc03AreaDescriptorSubtypeEnum** (by restriction)

Sub-types: None

Name TpegLoc03AreaDescriptorSubtypeEnum

Content

- Base XSD Type: string
- *value* comes from list: {'administrativeAreaName'|'administrativeReferenceName'|'areaName'|'coun

Documentation

Descriptors for describing area locations.

Schema Component Representation

```
<xs:simpleType name="TpegLoc03AreaDescriptorSubtypeEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="administrativeAreaName"/>
  </xs:restriction>
</xs:simpleType>
```

```

<xs:enumeration value="administrativeReferenceName"/>
<xs:enumeration value="areaName"/>
<xs:enumeration value="countyName"/>
<xs:enumeration value="lakeName"/>
<xs:enumeration value="nationName"/>
<xs:enumeration value="policeForceControlAreaName"/>
<xs:enumeration value="regionName"/>
<xs:enumeration value="seaName"/>
<xs:enumeration value="townName"/>
<xs:enumeration value="other"/>
</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'|'are

Documentation Descriptors other than junction names and road descriptors which can help to identify t

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: **TpegLoc04HeightTypeEnum**

Super-types: [xs:string](#) < **TpegLoc04HeightTypeEnum** (by restriction)

Sub-types: None

Name TpegLoc04HeightTypeEnum

Content

- Base XSD Type: string
- *value* comes from list:
{'above'|'aboveSeaLevel'|'aboveStreetLevel'|'at'|'atSeaLevel'|'atStreetLevel'|'l

Documentation Types of height.

Schema Component Representation

```
<xs:simpleType name="TpegLoc04HeightTypeEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="above"/>
    <xs:enumeration value="aboveSeaLevel"/>
    <xs:enumeration value="aboveStreetLevel"/>
    <xs:enumeration value="at"/>
    <xs:enumeration value="atSeaLevel"/>
    <xs:enumeration value="atStreetLevel"/>
    <xs:enumeration value="below"/>
    <xs:enumeration value="belowSeaLevel"/>
    <xs:enumeration value="belowStreetLevel"/>
    <xs:enumeration value="undefined"/>
    <xs:enumeration value="unknown"/>
    <xs:enumeration value="other"/>
  </xs:restriction>
</xs:simpleType>
```

[top](#)

Simple Type: **TruckParkingDynamicManagementEnum**

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

Sub-types: None

Name TruckParkingDynamicManagementEnum

Content

- Base XSD Type: string
- *value* comes from list:
{'compactParking'|'queueParking'|'noDynamicParkingManagement'|'other'}

Documentation Dynamic parking mode enum.

Schema Component Representation

```

<xs:simpleType name="TruckParkingDynamicManagementEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="compactParking"/>
    <xs:enumeration value="queueParking"/>
    <xs:enumeration value="noDynamicParkingManagement"/>
    <xs:enumeration value="other"/>
  </xs:restriction>
</xs:simpleType>

```

[top](#)

Simple Type: **UrbanParkingSiteTypeEnum**

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

Sub-types: None

Name UrbanParkingSiteTypeEnum

Content

- Base XSD Type: string
- *value* comes from list:
{'onStreetParking'|'offStreetParking'|'other'}

Documentation The type of an urban parking site.

Schema Component Representation

```

<xs:simpleType name="UrbanParkingSiteTypeEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="onStreetParking"/>
    <xs:enumeration value="offStreetParking"/>
    <xs:enumeration value="other"/>
  </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

Content

- Base XSD Type: anyURI

Documentation 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="Url">
  <xs:restriction base="xs:anyURI"/>
</xs:simpleType>

```

[top](#)

Simple Type: **UserTypeEnum**

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

Sub-types: None

Name UserTypeEnum

Content

- Base XSD Type: string
- *value* comes from list:
{'allUsers'|'hotelGuests'|'reservationHolders'|'registeredDisabledUsers'|'disabled'|'handicapped'|'hearingImpaired'|'visuallyImpaired'|'wheelchairUsers'|'men'|'women'}

Documentation Types of users; used for parking but also for usage of equipment and services.

Schema Component Representation

```

<xs:simpleType name="UserTypeEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="allUsers"/>
    <xs:enumeration value="hotelGuests"/>
    <xs:enumeration value="reservationHolders"/>
    <xs:enumeration value="registeredDisabledUsers"/>
    <xs:enumeration value="disabled"/>
    <xs:enumeration value="handicapped"/>
    <xs:enumeration value="hearingImpaired"/>
    <xs:enumeration value="visuallyImpaired"/>
    <xs:enumeration value="wheelchairUsers"/>
    <xs:enumeration value="men"/>
    <xs:enumeration value="women"/>
  </xs:restriction>
</xs:simpleType>

```

```

<xs:enumeration value="staff"/>
<xs:enumeration value="employees"/>
<xs:enumeration value="visitors"/>
<xs:enumeration value="overnightParker"/>
<xs:enumeration value="other"/>
<xs:enumeration value="unknown"/>
</xs:restriction>
</xs:simpleType>

```

[top](#)

Simple Type: **ValidityStatusEnum**

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

Sub-types: None

Name ValidityStatusEnum

Content

- Base XSD Type: string
- *value* comes from list:
{'active'|'suspended'|'definedByValidityTimeSpec'}

Documentation Values of validity status that can be assigned to a described event, action or item.

Schema Component Representation

```

<xs:simpleType name="ValidityStatusEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="active"/>
    <xs:enumeration value="suspended"/>
    <xs:enumeration value="definedByValidityTimeSpec"/>
  </xs:restriction>
</xs:simpleType>

```

[top](#)

Simple Type: **VehicleEquipmentEnum**

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

Sub-types: None

Name VehicleEquipmentEnum

Content

- Base XSD Type: string
- *value* comes from list:
{'notUsingSnowChains'|'notUsingSnowChainsOrTyres'|'snowChainsInUse'|'s

Documentation Types of vehicle equipment in use or on board.

Schema Component Representation

```

<xs:simpleType name="VehicleEquipmentEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="notUsingSnowChains"/>
    <xs:enumeration value="notUsingSnowChainsOrTyres"/>
  </xs:restriction>
</xs:simpleType>

```

```

<xs:enumeration value="snowChainsInUse"/>
<xs:enumeration value="snowTyresInUse"/>
<xs:enumeration value="snowChainsOrTyresInUse"/>
<xs:enumeration value="withoutSnowTyresOrChainsOnBoard"/>
</xs:restriction>
</xs:simpleType>

```

[top](#)

Simple Type: **VehicleType2Enum**

Super-types: [xs:string](#) < **VehicleType2Enum** (by restriction)

Sub-types: None

Name VehicleType2Enum

Content

- Base XSD Type: string
- *value* comes from list:
{motorhome|lightGoodsVehicle|heavyGoodsVehicle|minibus|smallCar|la

Documentation Vehicle types which are currently not supported in vehicleType.

Schema Component Representation

```

<xs:simpleType name="VehicleType2Enum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="motorhome"/>
    <xs:enumeration value="lightGoodsVehicle"/>
    <xs:enumeration value="heavyGoodsVehicle"/>
    <xs:enumeration value="minibus"/>
    <xs:enumeration value="smallCar"/>
    <xs:enumeration value="largeCar"/>
    <xs:enumeration value="lightGoodsVehicleWithTrailer"/>
    <xs:enumeration value="heavyGoodsVehicleWithTrailer"/>
    <xs:enumeration value="heavyHaulageVehicle"/>
    <xs:enumeration value="passengerCar"/>
    <xs:enumeration value="unknown"/>
  </xs:restriction>
</xs:simpleType>

```

[top](#)

Simple Type: **VehicleTypeEnum**

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

Sub-types: None

Name VehicleTypeEnum

Content

- Base XSD Type: string
- *value* comes from list:
{agriculturalVehicle|anyVehicle|articulatedVehicle|bicycle|bus|car|carav

Documentation Types of vehicle.

Schema Component Representation

```
<xs:simpleType name="VehicleTypeEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="agriculturalVehicle"/>
    <xs:enumeration value="anyVehicle"/>
    <xs:enumeration value="articulatedVehicle"/>
    <xs:enumeration value="bicycle"/>
    <xs:enumeration value="bus"/>
    <xs:enumeration value="car"/>
    <xs:enumeration value="caravan"/>
    <xs:enumeration value="carOrLightVehicle"/>
    <xs:enumeration value="carWithCaravan"/>
    <xs:enumeration value="carWithTrailer"/>
    <xs:enumeration value="constructionOrMaintenanceVehicle"/>
    <xs:enumeration value="fourWheelDrive"/>
    <xs:enumeration value="highSidedVehicle"/>
    <xs:enumeration value="lorry"/>
    <xs:enumeration value="moped"/>
    <xs:enumeration value="motorcycle"/>
    <xs:enumeration value="motorcycleWithSideCar"/>
    <xs:enumeration value="motorscooter"/>
    <xs:enumeration value="tanker"/>
    <xs:enumeration value="threeWheeledVehicle"/>
    <xs:enumeration value="trailer"/>
    <xs:enumeration value="tram"/>
    <xs:enumeration value="twoWheeledVehicle"/>
    <xs:enumeration value="van"/>
    <xs:enumeration value="vehicleWithCatalyticConverter"/>
    <xs:enumeration value="vehicleWithoutCatalyticConverter"/>
    <xs:enumeration value="vehicleWithCaravan"/>
    <xs:enumeration value="vehicleWithTrailer"/>
    <xs:enumeration value="withEvenNumberedRegistrationPlates"/>
    <xs:enumeration value="withOddNumberedRegistrationPlates"/>
    <xs:enumeration value="other"/>
  </xs:restriction>
</xs:simpleType>
```

[top](#)

Simple Type: **Volt**

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

Sub-types: None

Name Volt

Content

- Base XSD Type: float

Documentation Volt.

Schema Component Representation

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

[top](#)

Simple Type: **WeekOfMonthEnum**

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

Sub-types: None

Name WeekOfMonthEnum

Content

- Base XSD Type: string
- *value* comes from list:
{'firstWeekOfMonth'|'secondWeekOfMonth'|'thirdWeekOfMonth'|'fourthWeekOf'

Documentation Weeks of the month.

Schema Component Representation

```
<xs:simpleType name="WeekOfMonthEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="firstWeekOfMonth"/>
    <xs:enumeration value="secondWeekOfMonth"/>
    <xs:enumeration value="thirdWeekOfMonth"/>
    <xs:enumeration value="fourthWeekOfMonth"/>
    <xs:enumeration value="fifthWeekOfMonth"/>
  </xs:restriction>
</xs:simpleType>
```

[top](#)

Generated by [xs3p](#) ([old link](#)) . Last modified: 08/22/2016 10:41:30