

```

-----
-- POI from Projects SCOOP - INTERCOR - C-ROADS FR
--
-- Extension from ETSI TS 101 556-1 2012-07 (01v010101p(POI)) for charging spot
for electric vehicle (EVCSN Message definition)
--
-- This ASN.1 was generated: 13.08.2019 (DD/MM/YY) by the SCOOP - INTERCOR -
C-ROADS FR project
-- Published version location: Not defined yet, refer to handler of the projects
(FR)
-----

-----
-- ASN.1 Start Definition
-----

PoiManagement {cocsic (0) poi (101556) version (2)}
-- Replace : EVCSN-PDU-Descriptions {itu-t (0) identified-organization (4) etsi
(0) itsDomain (5) wg1 (1) ts (101556) evcsn (1) version (1)}

DEFINITIONS AUTOMATIC TAGS ::= BEGIN

IMPORTS

ItsPduHeader, StationID, TimestampIts, ReferencePosition
FROM ITS-Container {itu-t (0) identified-organization (4) etsi (0) itsDomain (5)
wg1 (1) ts (102894) cdd (2) version (1) }

Provider
FROM EfcDsrcApplication {iso(1) standard(0) 14906 application(0) version5(5)};

-----
-- POIType Section
-- Used as well by EVCSN (ETSI TS 101 556-1 2012-07) than by POI Type introduced
by FR projects (this asn extension)
-----

POIType ::= INTEGER(0..65535)
-- Value "1" is for EVCS (as specified in ETSI TS 101 556-1 2012-07). Other
value 2 to 65535 are used for other POI type (FR projects) with 11 for slots
(SlotReferenceStatusPdu structure), 12 for docks (DockTimeslotPdu structure),
7522 for ParkAndRide (BasicPoi structure) or 20028 for CoachAndLorryParking
(BasicPoi structure)..

ItsPOIHeader ::= SEQUENCE {
    poiType          POIType, -- see previous comment
    timeStamp        TimestampIts,
    relayCapable      BOOLEAN -- set to "false" when not relevant for the
poiType
}
-----

```

-- Common Section
-- Used by several section

OpeningPeriod ::= SEQUENCE {
 opening TimeOfDay, -- Number of minutes after 00:00
when the parking opens or start of timeslot
 closing TimeOfDay -- Number of minutes after 00:00
when the parking closes or end of timeslot
}

TimeOfDay ::= INTEGER {
 midnight(0),
 oneMinuteAfterMidnight(1)
} (0..1440)

-- EVCSN Data Section
-- This part is unchanged in reference to the standard ETSI TS 101 556 2012-07.
-- This part is used when POIType will be "1" in the entry.
-- For other services (POIType from 2 to 65535), introduced by this asn
extension, the following part is not used (skip to next section).

EvcsnPdu ::= SEQUENCE {
 header ItsPduHeader,
 evcsn EVChargingSpotNotificationPOIMessage
}

EVChargingSpotNotificationPOIMessage ::= SEQUENCE {
 poiHeader ItsPOIHeader, -- Specific POI Message Header
 evcsnData ItsEVCSNData -- Electric Vehicle Charging Spot Data Elements
}

ItsEVCSNData ::= SEQUENCE {
 totalNumberOfStations NumberStations,
 chargingStationsData SEQUENCE (SIZE(1..256)) OF ItsChargingStationData
}

ItsChargingStationData ::= SEQUENCE {
 chargingStationID StationID,
 utilityDistributorId UTF8String (SIZE(1..32)) OPTIONAL,
 providerID UTF8String (SIZE(1..32)) OPTIONAL,
 chargingStationLocation ReferencePosition,
 address UTF8String OPTIONAL,
 phoneNumber NumericString (SIZE(1..16)) OPTIONAL,
 accessibility UTF8String (SIZE(1..32)),
 digitalMap DigitalMap OPTIONAL,
 openingDaysHours UTF8String,
 pricing UTF8String,
 bookingContactInfo UTF8String OPTIONAL,
 payment UTF8String OPTIONAL,
 chargingSpotsAvailable ItsChargingSpots,

```

    ...
}

ItsChargingSpots ::= SEQUENCE (SIZE(1..16)) OF ItsChargingSpotDataElements

ItsChargingSpotDataElements ::= SEQUENCE {
    type                ChargingSpotType,
    evEquipmentID       UTF8String OPTIONAL,
    typeOfReceptacle    TypeOfReceptacle,
    energyAvailability   UTF8String,
    parkingPlacesData   ParkingPlacesData OPTIONAL
}

DigitalMap ::= SEQUENCE (SIZE(1..256)) OF ReferencePosition

ChargingSpotType ::= BIT STRING {
    standardChargeMode1(0),
    standardChargeMode2(1),
    standardOrFastChargeMode3(2),
    fastChargeWithExternalCharger(3),
    quickDrop(8),
    inductiveChargeWhileStationary(12),
    inductiveChargeWhileDriving(14)
}

TypeOfReceptacle ::= BIT STRING

ParkingPlacesData ::= SEQUENCE (SIZE(1..4)) OF SpotAvailability

SpotAvailability ::= SEQUENCE {
    maxWaitingTimeMinutes INTEGER (0..1400), -- 0 if available or max waiting
time (minutes)
    blocking                BOOLEAN          -- true if the spot can be blocked
}

NumberStations ::= INTEGER(1..256)

-----
-- BasicPoi Data Section
-- This part is an extension of the asn of TS 101 556 in reference to the
standard.
-- This part is used when POIType will be e.g "7520" ParkingLot (free park) or
"7521" ParkingGarage (not free) or "7522" ParkAndRide or "20028"
CoachandLorryParking.
-----

BasicPoiPdu ::= SEQUENCE {
    header        ItsPduHeader,
    basicPoi      BasicPoiMessage
}

BasicPoiMessage ::= SEQUENCE {
    poiHeader      ItsPOIHeader,

```

```

        poiNumber      PoiNumber,
        location        LocationData,
        status          StatusData
    }

PoiNumber ::= SEQUENCE {
    serviceProviderId      Provider, -- Imported from 14906 (IVI)
    basicPoiNumber          BasicPoiNumber -- Unique identifier of
the POI service for the provider
}

BasicPoiNumber ::= INTEGER(0..65535)

LocationData ::= SEQUENCE {
    refPoint              ReferencePosition,
    name                  UTF8String (SIZE (1..31)),
    address                UTF8String (SIZE
(1..255)) OPTIONAL,
    phoneNumber            UTF8String (SIZE (1..31)) OPTIONAL, --
eg +33 1 23 45 67 89
    website                UTF8String (SIZE (1..31))
OPTIONAL -- without "http://"
}

StatusData ::= SEQUENCE {
    openingStatus          OpeningStatus,
    parkingStatus          ParkingStatus,
    openingDaysHours       SEQUENCE (SIZE (7)) OF DailyOpeningHours
OPTIONAL, -- Opening hours by day, starting by monday, in local time.
    additionalInformation  UTF8String (SIZE (1..255)) OPTIONAL
}

OpeningStatus ::= INTEGER {
    closed                 (0),
    open                   (1),
    subscriberonly         (2),
    unknown                 (15)
} (0..15)

ParkingStatus ::= SEQUENCE {
    freeSpots              FreeSpots,
    totalSpots              TotalSpots
}

FreeSpots ::= INTEGER {
    full                   (0),
    onespot                (1), -- only one place is available
    freespaces              (16382),
    unknown                 (16383)
} (0..16383)

TotalSpots ::= INTEGER {
    noparking              (0),
    unknown                 (16383)
}

```

}(0..16383)

DailyOpeningHours ::= SEQUENCE (SIZE(0..4)) OF OpeningPeriod

-- SlotReferenceStatus Data Section
-- This part is an extension of the asn of TS 101 556 in reference to the
standard.
-- This part is used when POIType will be "11".

SlotReferenceStatusPdu ::= SEQUENCE {
 header ItsPduHeader,
 data SlotReferenceStatusMessage
}

SlotReferenceStatusMessage ::= SEQUENCE {
 poiHeader ItsPOIHeader,
 data SlotReferenceStatusData
}

SlotReferenceStatusData ::= SEQUENCE {
 port UTF8String (SIZE (5)), -- Locode Geographical
place where the slot is validated
 operator UTF8String (SIZE (1..31)), -- Operator code: Ex
: APPLUSDKE
 terminal UTF8String (SIZE (1..50)), -- SIRET of terminal
for which the status are given
 slotReference UTF8String (SIZE (1..50)), -- Booking reference for
pick-up and or Cargo reference for delivery
 equipmentId UTF8String (SIZE (1..31)) OPTIONAL, -- Container
number
 operationType INTEGER(0..399), -- Trade and Transport status codes
Revision 6, UNECE CEFAC Trade Facilitation Recommendation No. 24
 expirationDate TimestampIts -- Date of expiry of the status
}

-- DockTimeslot Data Section
-- This part is an extension of the asn of TS 101 556 in reference to the
standard.
-- This part is used when POIType will be "12".

DockTimeslotPdu ::= SEQUENCE {
 header ItsPduHeader,
 data DockTimeslotMessage
}

DockTimeslotMessage ::= SEQUENCE {
 poiHeader ItsPOIHeader,
 provider Provider,

```

        terminalName    UTF8String (SIZE (1..31)), -- Name of Terminal
        webServiceUrl   UTF8String, -- WebService URL to do the dock and
timeslot reservation. Format http(s)://xxx.yy.com
        docks           SEQUENCE (SIZE (1..31)) OF
DockTimeslotInformation
}

DockTimeslotInformation ::= SEQUENCE {
    dockId              INTEGER (1..16383), -- Internal
Id of dock in the terminal
    dockName            UTF8String (SIZE (1..31)), --
dock name
    availableDayTimeslots SEQUENCE (SIZE (0..7)) OF DayTimeslotInformation
-- List of days of available timeslots
}

DayTimeslotInformation ::= SEQUENCE {
    dayTimeStamp        TimestampIts,-- Available day
    availableTimeslots  SEQUENCE (SIZE (0..23)) OF OpeningPeriod -- List
of available timeslots of the day
}

END

-- ASN.1 End

```