PD ISO/TS 21219-25:2017

BSI Standards Publication

Intelligent transport systems — Traffic and travel information (TTI) via transport protocol experts group, generation 2 (TPEG2)

Part 25: Electromobility charging infrastructure (TPEG2-EMI)

bsi.

National foreword

This Published Document is the UK implementation of ISO/TS 21219-25:2017.

The UK participation in its preparation was entrusted to Technical Committee EPL/278, Intelligent transport systems.

A list of organizations represented on this committee can be obtained on request to its secretary.

This publication does not purport to include all the necessary provisions of a contract. Users are responsible for its correct application.

Published by BSI Standards Limited 2017

ISBN 978 0 580 93252 6
ICS 03.220.01; 35.240.60

Compliance with a British Standard cannot confer immunity from legal obligations.

This Published Document was published under the authority of the Standards Policy and Strategy Committee on 28 February 2017.

Amendments/corrigenda issued since publication

Date	Text affected

PD ISO/TS 21219-25:2017

TECHNICAL SPECIFICATION

ISO/TS 21219-25

First edition
2017-02

Intelligent transport systems — Traffic and travel information (TTI) via transport protocol experts group, generation 2 (TPEG2) —

Part 25: Electromobility charging infrastructure (TPEG2-EMI)

Systèmes intelligents de transport — Informations sur le trafic et le tourisme via le groupe expert du protocole de transport, génération 2 (TPEG2) —

Partie 25: Infrastructure pour l'alimentation en électromobilité (TPEG2-EMI)

Reference number
ISO/TS 21219-25:2017(E)

ISO copyright office
Ch. de Blandonnet 8 • CP 401
CH-1214 Vernier, Geneva, Switzerland
Tel. +41 22 749 01 11
Fax +41 22 749 09 47
copyright@iso.org
www.iso.org

Contents

Page

Foreword

ISO (the International Organization for Standardization) is a worldwide federation of national standards bodies (ISO member bodies). The work of preparing International Standards is normally carried out through ISO technical committees. Each member body interested in a subject for which a technical committee has been established has the right to be represented on that committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work. ISO collaborates closely with the International Electrotechnical Commission (IEC) on all matters of electrotechnical standardization.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular, the different approval criteria needed for the different types of ISO documents should be noted. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see www.iso.org/directives).

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO shall not be held responsible for identifying any or all such patent rights. Details of any patent rights identified during the development of the document will be in the Introduction and/or on the ISO list of patent declarations received (see www.iso.org/patents).

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation on the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the World Trade Organization (WTO) principles in the Technical Barriers to Trade (TBT) see the following URL: www.iso.org/iso/foreword.html.

The committee responsible for this document is ISO/TC 204, *Intelligent transport systems*.

A list of all parts in the ISO/TS 21219 series can be found on the ISO website.

Introduction

History

TPEG technology was originally proposed by the European Broadcasting Union (EBU) Broadcast Management Committee, who established the B/TPEG project group in the autumn of 1997 with a brief to develop, as soon as possible, a new protocol for broadcasting traffic and travel-related information in the multimedia environment. TPEG technology, its applications and service features were designed to enable travel-related messages to be coded, decoded, filtered and understood by humans (visually and/or audibly in the user's language) and by agent systems. Originally, a byte-oriented data stream format, which may be carried on almost any digital bearer with an appropriate adaptation layer, was developed. Hierarchically structured TPEG messages from service providers to end-users were designed to transfer information from the service provider database to an end-user's equipment.

One year later, in December 1998, the B/TPEG group produced its first EBU specifications. Two documents were released. Part 2 (TPEG-SSF, which became ISO/TS 18234-2) described the Syntax, Semantics and Framing structure, which was used for all TPEG applications. Meanwhile, Part 4 (TPEG-RTM, which became ISO/TS 18234-4) described the first application for Road Traffic Messages.

Subsequently, in March 1999, CEN/TC 278, in conjunction with ISO/TC 204, established a group comprising members of the former EBU B/TPEG and this working group continued development work. Further parts were developed to make the initial set of four parts enabling the implementation of a consistent service. Part 3 (TPEG-SNI, ISO/TS 18234-3) described the Service and Network Information Application used by all service implementations to ensure appropriate referencing from one service source to another.

Part 1 (TPEG-INV, ISO/TS 18234-1) completed the series by describing the other parts and their relationship; it also contained the application IDs used within the other parts. Additionally, Part 5, the Public Transport Information Application (TPEG-PTI, ISO/TS 18234-5), was developed. The so-called TPEG-LOC Location Referencing method, which enabled both map-based TPEG-decoders and non-map-based ones to deliver either map-based Location Referencing or human readable text information, was issued as ISO/TS 18234-6 to be used in association with the other applications parts of the ISO/TS 18234 series to provide location referencing.

The ISO/TS 18234 series has become known as TPEG Generation 1.

TPEG Generation 2

When the Traveller Information Services Association (TISA), derived from former forums, was inaugurated in December 2007, TPEG development was taken over by TISA and continued in the TPEG applications working group.

It was about this time that the (then) new Unified Modelling Language (UML) was seen as having major advantages for the development of new TPEG Applications in communities who would not necessarily have binary physical format skills required to extend the original TPEG TS work. It was also realized that the XML format for TPEG described within the ISO/TS 24530 series (now superseded) had a greater significance than previously foreseen, especially in the content-generation segment and that keeping two physical formats in synchronism, in different standards series, would be rather difficult.

As a result, TISA set about the development of a new TPEG structure that would be UML based. This has subsequently become known as TPEG Generation 2.

TPEG2 is embodied in the ISO/TS 21219 series and it comprises many parts that cover introduction, rules, toolkit and application components. TPEG2 is built around UML modelling and has a core of rules that contain the modelling strategy covered in ISO/TS 21219-2, ISO/TS 21219-3, ISO/TS 21219-4 and the conversion to two current physical formats: binary and XML; others could be added in the future. TISA uses an automated tool to convert from the agreed UML model XMI file directly into an MS Word document file, to minimize drafting errors, that forms the annex for each physical format.

TPEG2 has a three container conceptual structure: Message Management (ISO/TS 21219-6), Application (several parts) and Location Referencing (ISO/TS 21219-7[1)]). This structure has flexible capability and can accommodate many differing use cases that have been proposed within the TTI sector and wider for hierarchical message content.

TPEG2 also has many location referencing options as required by the service provider community, any of which may be delivered by vectoring data included in the Location Referencing Container.

The following classification provides a helpful grouping of the different TPEG2 parts according to their intended purpose.

— Toolkit parts: TPEG2-INV (ISO/TS 21219-1), TPEG2-UML (ISO/TS 21219-2), TPEG2-UBCR (ISO/TS 21219-3), TPEG2-UXCR (ISO/TS 21219-4), TPEG2-SFW (ISO/TS 21219-5), TPEG2-MMC (ISO/TS 21219-6), TPEG2-LRC (ISO/TS 21219-7), TPEG2-LTE (ISO/TS 21219-24);

— Special applications: TPEG2-SNI (ISO/TS 21219-9), TPEG2-CAI (ISO/TS 21219-10);

— Location Referencing: TPEG2-ULR (ISO/TS 21219-11[2)]), TPEG2-GLR (ISO/TS 21219-21[3)]), TPEG2-OLR (ISO/TS 21219-22[4)]);

— Applications: TPEG2-PKI (ISO/TS 21219-14), TPEG2-TEC (ISO/TS 21219-15), TPEG2-FPI (ISO/TS 21219-16), TPEG2-TFP (ISO/TS 21219-18), TPEG2-WEA (ISO/TS 21219-19), TPEG2-RMR (ISO/TS 21219-23), TPEG2-EMI (ISO/TS 21219-25).

TPEG2 has been developed to be broadly (but not totally) backward compatible with TPEG1 to assist in transitions from earlier implementations, while not hindering the TPEG2 innovative approach and being able to support many new features, such as dealing with applications having both long-term, unchanging content and highly dynamic content, such as Parking Information.

This document is based on the TISA specification technical/editorial version reference:

SP13010/1.0/001

1) Under development.

2) Under development.

3) Under development.

4) Under development.

Intelligent transport systems — Traffic and travel information (TTI) via transport protocol experts group, generation 2 (TPEG2) —

Part 25: Electromobility charging infrastructure (TPEG2-EMI)

1 Scope

This document defines the TPEG application electromobility charging infrastructure (EMI). It has been specifically designed to support information about charging infrastructure for electric vehicles (not just cars), the location of e-charging points and their suitability for the respective vehicle (e.g. connector type, charging modality). As electric vehicles will occupy a "charging space" for a longer period of time, information on availability/waiting time and reservation options are highly relevant for a user of an electric vehicle to optimally plan his route/trip and are therefore also accounted for.

The standardized delivery, through a TPEG technology, of information on charging infrastructures has the following benefits to an end user of this TPEG service:

a) Identifying suitable charging units for his vehicle, thus preventing unnecessary driving around to find a fitting unit (also has environmental benefits).

b) Verifying the real-time availability of charging units.

c) Being able to plan ahead and reserve a spot in a charging park and thus optimize the planning of his trip.

d) Being able to select a financially attractive charging point in a charging park the operator of which has billing agreements with the user's electromobility provider.

In addition to these end-user benefits, also electromobility providers and charging park operators benefit from a standardized TPEG format as it allows an easier harmonization of the electromobility charging infrastructure information with the data formats used for the exchange of information between management systems of electromobility providers and charge park operators and according specifications (e.g. Open Charge Alliance[5], eMobility ICT Interoperability Innovation (eMI[3])[6], etc.).

The TPEG application electromobility charging infrastructure, as add-on service component next to, for example traffic information, is laid out to support large numbers of charge parks with only modest bandwidth requirements.

2 Normative references

The following documents are referred to in text in such a way that some or all of their content constitutes requirements of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO/TS 18234-11:2013, *Intelligent transport systems — Traffic and Travel Information (TTI) via transport protocol experts group, generation 1 (TPEG1) binary data format — Part 11: Location Referencing Container (TPEG1-LRC)*

5) http://www.openchargealliance.org/

6) http://emi3group.com/

ISO/TS 21219-1, *Intelligent transport systems — Traffic and travel information (TTI) via transport protocol experts group, generation 2 (TPEG2) — Part 1: Introduction, numbering and versions (TPEG2-INV)*

ISO/TS 21219-5, *Intelligent transport systems — Traffic and travel information (TTI) via transport protocol experts group, generation 2 (TPEG2) — Part 5: Service framework (TPEG2-SFW)*

ISO/TS 21219-6, *Intelligent transport systems — Traffic and travel information (TTI) via transport protocol experts group, generation 2 (TPEG2) — Part 6: Message management container (TPEG2-MMC)*

ISO/TS 21219-9, *Intelligent transport systems — Traffic and travel information (TTI) via transport protocol experts group, generation 2 (TPEG2) — Part 9: Service and network information (TPEG2-SNI)*

3 Terms and definitions

For the purposes of this document, the following terms and definitions apply.

ISO and IEC maintain terminological databases for use in standardization at the following addresses:

— IEC Electropedia: available at http://www.electropedia.org/

— ISO Online browsing platform: available at http://www.iso.org/obp

3.1
electric vehicle
EV
vehicle that is (partly) electrically powered and operated

Note 1 to entry: With respect to the TPEG requirements, other electric vehicles such as e-bikes are considered.

Note 2 to entry: EV batteries can typically be charged at any regular power socket. In case fast charging is required, e.g. during longer journeys, higher demands on the technical infrastructure are made. Specific sockets and high-power connector cables have been developed to allow a simple and secure usage of boost *charging stations* (3.3). It is necessary for the end user to know which options are supported by a charging station. EVs may have an "identity" for electronic readout, e.g. by means of a certificate. Also, other information which can be communicated by an EV to the infrastructure may be relevant for the execution and planning of charging orders. The current battery charge condition, the power requirements during the charging procedure, as well as the cruising range are parameters that may be relevant for the planning of charging orders. The vehicle and charging station can communicate using a connector cable, but also other mechanisms are possible, e.g. using the back-end system of an EV manufacturer, to which an EV is connected (using mobile data connection).

3.2
charging park
park that consists of multiple physical *charging stations* (3.3) which technically and/or logically belong together and are being operated together

Note 1 to entry: This may be the case, for example, in a commercially operated car park or in a city district where publicly operated charging stations are grouped together. Charging parks are being operated by *charging park operators* (3.7).

3.3
charging station
station that consists of a physical unit (typically a column or cabinet-like structure) containing and managing one or more *charging points* (3.4) offering the end-user the possibility to be authorized (typically by means of a card reader) and activate one of the charging points at the charging station, hook up the *electric vehicle* (3.1) and start the charging procedure

3.4
charging point
unit in a *charging station* (3.3) at which an *electric vehicle* (3.1) can be supplied with power

Note 1 to entry: A charging station can provide multiple charging points, which again can contain multiple sockets to support more than one charging connector type. In general, as soon as one socket at a charging point is in use, the charging point is occupied. Typically for each charging point, a parking space is provided at the charging station.

3.5
energy provider
provider that includes all relevant energy suppliers, local solar power generators, as well as traditional major companies in the power industry sector

Note 1 to entry: It is their role to provide energy to the *charging park operators* (3.7).

3.6
electromobility provider
EM provider
business partner for the end-users who charge their *electric vehicles* (3.1)

Note 1 to entry: Typically, an end-user has a contract with an EM provider, the details of which are connected to an (RF) ID card [having an (internationally) unique card number] that is used for authorization and billing. The EM provider sees to it that his customers can charge their vehicles in as many *charging parks* (3.2) as possible and bills the customer according to the respective contract.

3.7
charging park operator
operator who manages one or more *charging parks* (3.2)

Note 1 to entry: The charging park operator maintains the charging site(s) and is a business partner to the energy provider and the *EM provider* (3.6). Typically, a charging park operator bills the end-user based on "roaming agreements" with multiple EM providers.

3.8
electric vehicle supply equipment identity
EVSEID
ID that uniquely identifies a concrete *charging point* (3.4) globally

Note 1 to entry: If a *charging station* (3.3) has multiple charging points, multiple EVSEIDs are used. See also DIN SPEC 91286.

4 Abbreviated terms

ACID	Application and Content Identifier
ADC	Application Data Container
CEN	Comité Européen de Normalization
EBU	European Broadcasting Union
EM	ElectroMobility
EMI	ElectroMobility Charging Infrastructure
EV	Electric Vehicle
EVSE	Electric Vehicle Supply Equipment
EVSEID	Electric Vehicle Supply Equipment Identity

LRC	Location Referencing Container
MMC	Message Management Container
OSI	Open Systems Interconnection
SFW	TPEG Service Framework: Modelling and Conversion Rules
TISA	Traveller Information Services Association
TPEG	Transport Protocol Expert Group
TTI	Traffic and Traveller Information
UML	Unified Modelling Language

5 Application specific constraints

5.1 Application identification

The word “application” is used in the TPEG specifications to describe specific subsets of the TPEG structure. An application defines a limited vocabulary for a certain type of messages, for example, parking information or road traffic information. Each TPEG application is assigned a unique number, called the Application IDentification (AID). An AID is defined whenever a new application is developed and these are all listed in ISO/TS 21219-1.

The application identification number is used within the TPEG2-SNI application ISO/TS 21219-9 to indicate how to process TPEG content and facilitates the routing of information to the appropriate application decoder.

5.2 Version number signalling

Version numbering is used to track the separate versions of an application through its development and deployment. The differences between these versions can have an impact on client devices.

The version numbering principle is defined in ISO/TS 21219-1.

Table 1 shows the current version numbers for signalling EMI within the SNI application ISO/TS 21219-9.

Table 1 — Current version numbers for signalling of EMI

major version number	1
minor version number	0

5.3 Ordered components

TPEG2-EMI requires a fixed order of TPEG components. The order for the EMI message component is shown in Figure 1; the first component shall be the *Message Management Container.* This shall be the only component if the message is a cancellation message. Otherwise, the MMC component shall be followed by the one or more *Application Data Container* component(s) which includes the application-specific information.

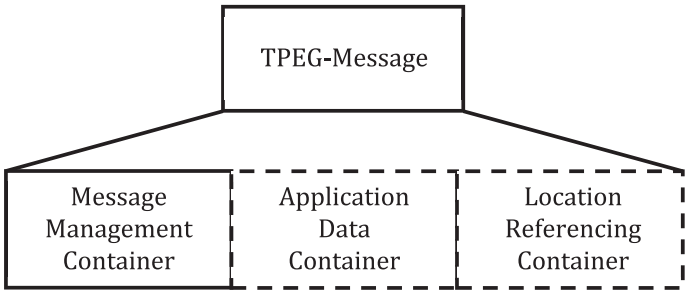

Figure 1 — Composition of TPEG messages

5.4 Extension

The requirement of a fixed component order does not affect the extension of EMI. Future application extensions may insert new components or may replace existing components by new ones without losing backward compatibility. An EMI decoder shall be able to detect and skip unknown components.

5.5 TPEG Service Component Frame

EMI makes use of the "Service Component Frame with dataCRC and messageCount" according to ISO/TS 21219-5 (TPEG2-SFW).

6 EMI structure

6.1 Overview

In Clause 6, the main structure of EMI and its capabilities are explained.

The EMI design is based on a distinction between information with a generally static reference style with an expected low refresh rate, and information of a more dynamic nature status with an expected high refresh rate.

6.2 EMI structuring considerations

6.2.1 Information aggregation level: Charging parks, charging stations, charging points

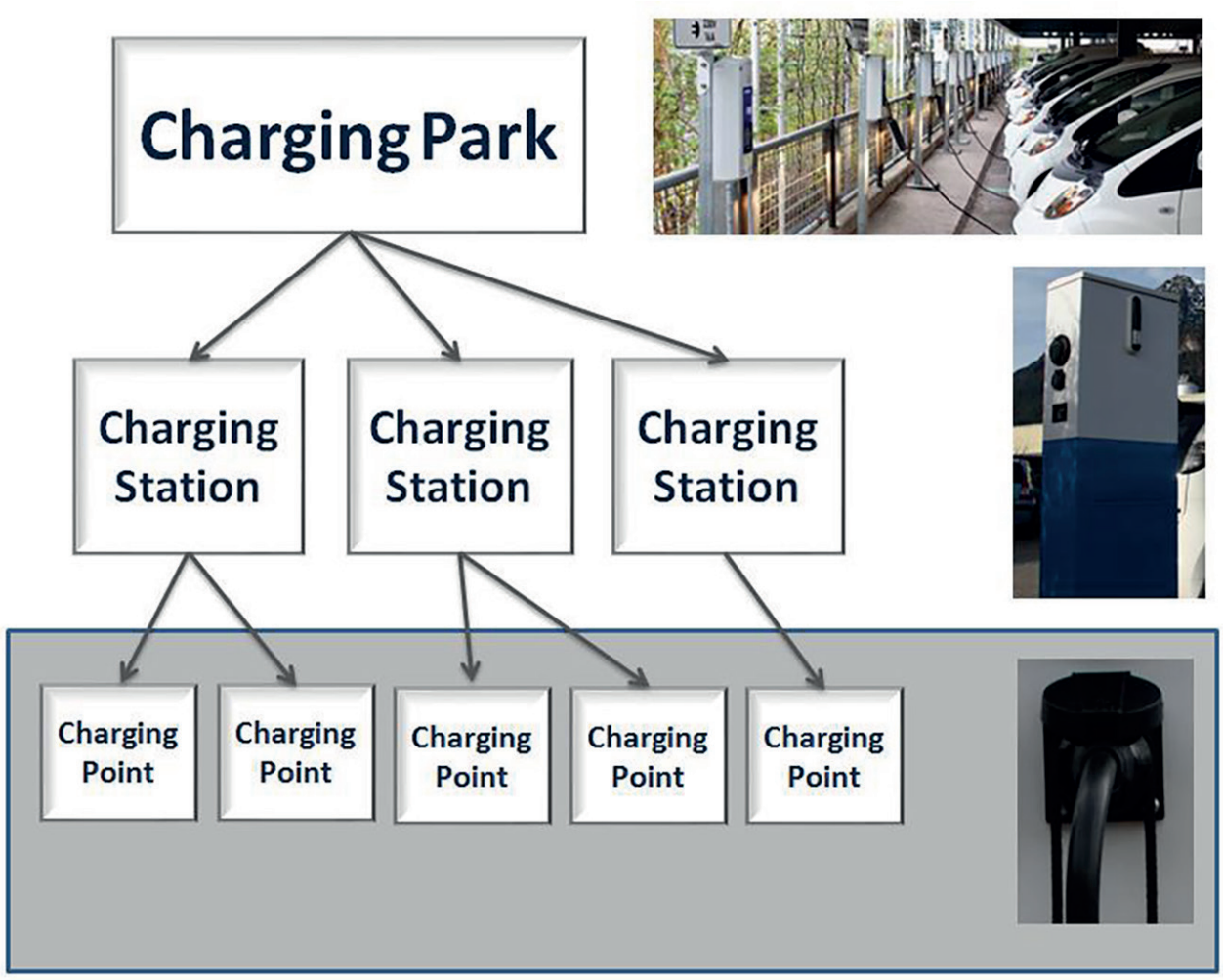

Figure 2 — EMI Structure

EMI (see structure in Figure 2) must provide an end user with enough information to find a suitable and available charging point to charge his electric vehicle. From an end user point of view, it is sufficient to obtain information at the level of a charging station: the physical location of a charging station matching user needs by providing sufficient information. It is not relevant for the user to know e.g. which physical charging point at a charging station would be available.

Therefore, in EMI information is aggregated either at charging park or charging station level. By doing so, EMI also takes into account that the number of charging points within a TPEG service may become too high to transmit all descriptive data as part of the general service. Thus, this approach supports efficient use of the transmission channel.

Detailed information on a specific charging point may become relevant to an end user wanting to make a reservation at a specific charging station. Therefore, in addition to the information provisioning on charging parks and charging stations, EMI supports a request and response session, to allow a reservation for a specific charging point at a given charging station (or in a charging park, respectively); see 6.2.3.

6.2.2 Static vs. dynamic information: Charging park information, charging park availability

An EMI service provider needs to be able to provide a TPEG client with a large amount of data at a relatively low transmission data rate. The typical TPEG concept, in which a single TPEG message equates with a single content item, cannot be applied for EMI, as it would take too much time to provide clients without any pre-existing information (e.g. transit users) with some useable data. Some form of transmission at high repetition rates for minimum content, augmented with low repetition rate for additional detailed content is then required.

Moreover, EMI contains information that are generally static (see Figure 3, typically descriptive information on charging parks) and information that may be updated frequently (see Figure 4, such as the availability information). EMI must also consider this information quality to support different repetition rates.

EMI has been designed to allow service providers to arrange their transmissions flexibly, depending on the volume of data to be transmitted and the available data rate. A TPEG message may contain partial or complete content for a charging park/charging station. A service provider may choose to aggregate descriptive information at the level of charging park in case only limited data rate is available. This typically static information shall be combined with location information.

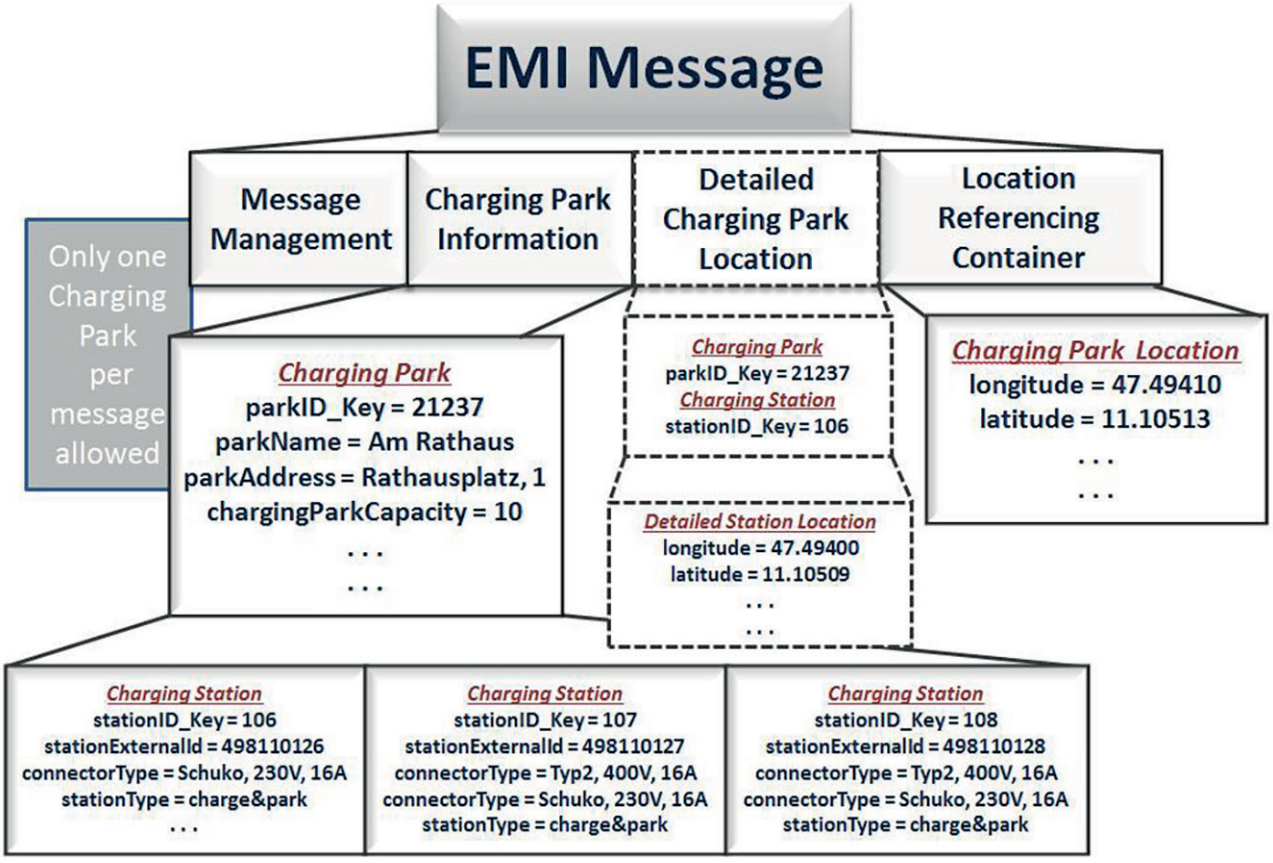

Figure 3 — Static information (example)

The availability information for charging parks/charging stations (typically highly dynamic information) is contained in separate data structures, which have been kept limited and compact to allow it to be transmitted with a higher refresh rate than the descriptive information. TPEG messages containing availability information shall not contain location information. Based on respective ID keys the TPEG client can recombine the availability information with the other information for charging parks and charging stations.

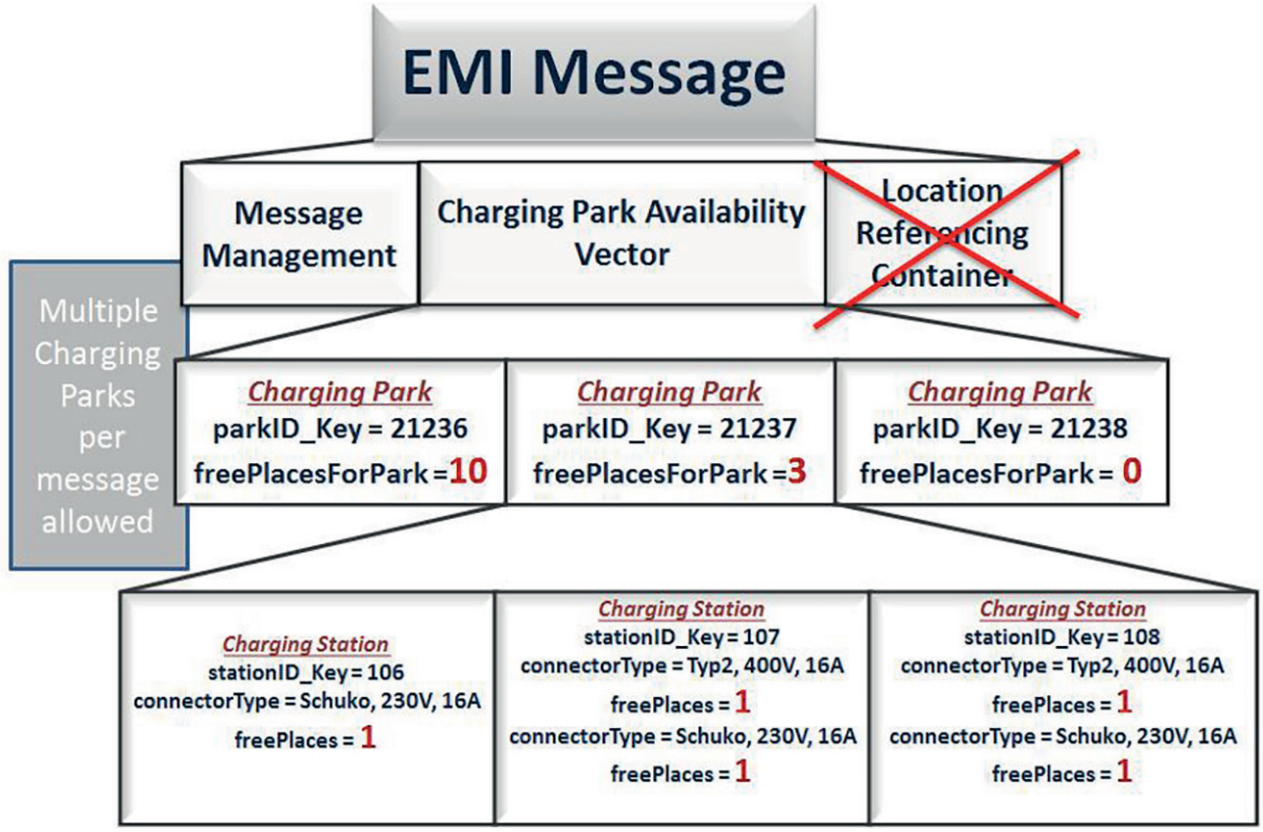

Figure 4 — Dynamic information (example)

6.2.3 Request and response

EMI supports a request and response session, to request the reservation of a ChargingPoint that matches the end user's requirements and the vehicle's attributes and to convey the required parameters to a reservation service (typically offered by an EM provider).

A sample EMI request message is shown in Figure 5. In the EMI request, either the charging station must be referenced directly or alternatively at a point near which an available charging station is searched. One of three options for this location referencing shall be included in the request message:

— internal TPEG reference (parkID_Key and optionally stationID_Key);

— external reference (stationExternalID, e.g. EVSE ID);

— coordinates of a point (longitude, latitude) and optionally parkOperator and/or providerExternalID.

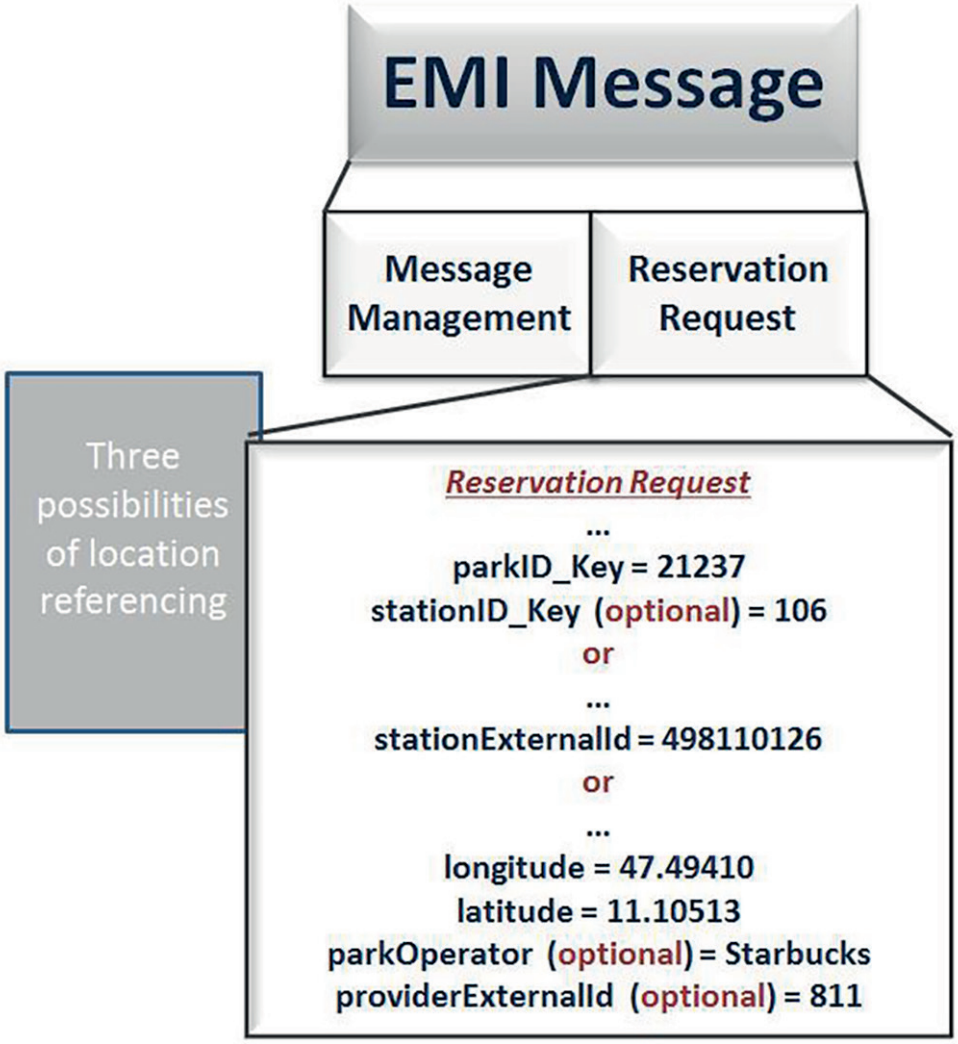

Figure 5 — Reservation request (example)

A sample EMI response message is shown in Figure 6. Use of the reservationConfirmed attribute indicates whether or not the reservation request is confirmed. In the case where the request is not confirmed, the reservationFreeText can be used to give additional information. In the case where the request is confirmed it is recommended to use venueExternalID (to address either a charging station or a charging point) and reservationID to unambiguously identify the reservation.

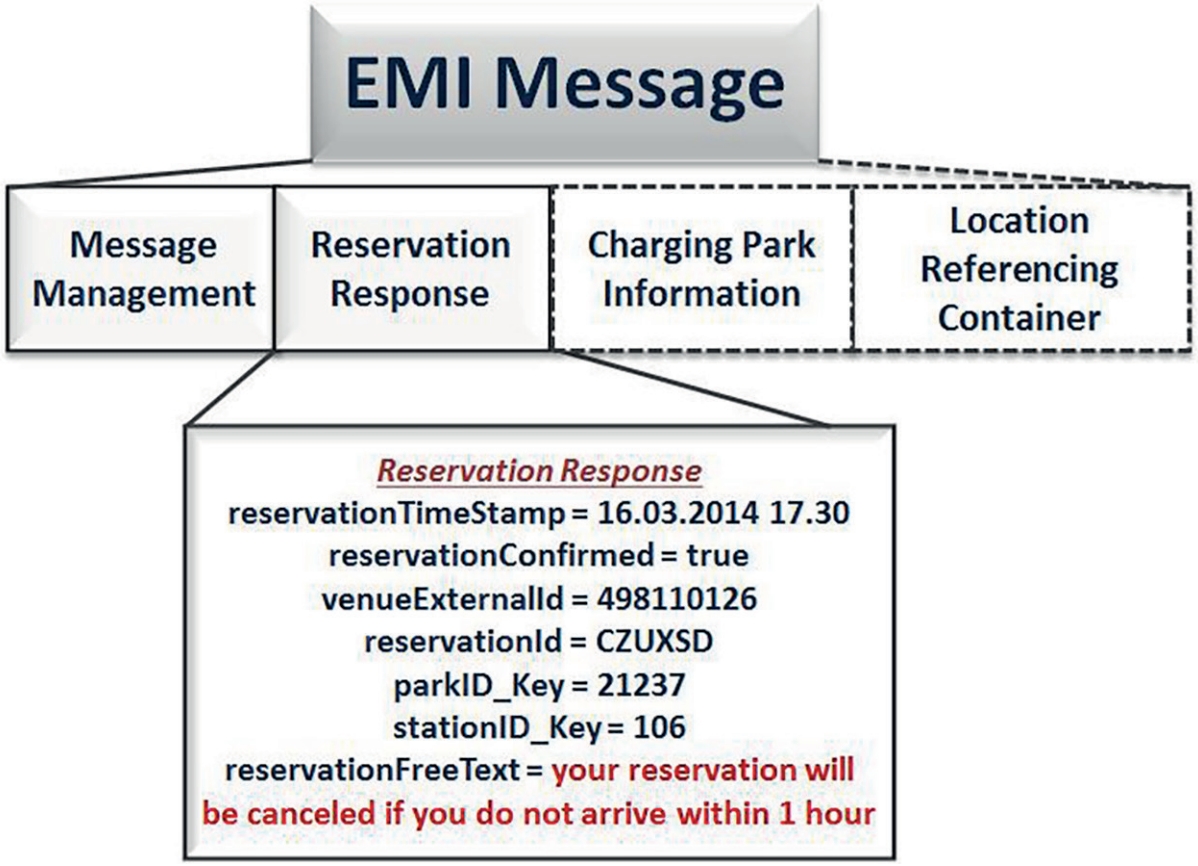

Figure 6 — Reservation response (example)

6.3 Pricing information

The pricing and billing models in the current electromobility market place are very different from fuel pricing/billing and currently also differ very much from one to the other. This mainly is caused by the many business actors in the "habitat" related to charging procedures, their relationships and business models.

The end-user typically has a business relationship with an EM provider, by means of a contract, based on which the end user is billed. This contract is somewhat comparable to a mobile phone contract, can take many different forms [e.g. prepaid (possibly differentiating rates based on vehicle type or amount of power used, …), flat rate or other].

The charging park operator or infrastructure provider in such case typically requires an end user to identify/authorize himself by means of the customer card issued by his EM provider. The charging park operator does not bill the end user, but rather gets paid by the EM provider based on "roaming agreements" (such "roaming" allows end users to make use of the charging infrastructure independent of the actual provider on-site). The infrastructure and power provider have business relationships with the EM provider (typically more than one), in which wholesale prices (rather than end user prices) are set. Thus, these wholesale agreements on pricing are transparent to the end user; for him only the contract with the respective EM provider counts.

Hence, for the end user is most important that he knows that with the customer card from his EM provider he can identify and authorize himself at a charging station, use it and be billed.

Moreover, the charging park operator is oblivious to the agreements an end user made with his EM provider and hence cannot make any reasonable assumption as to which costs will arise for an end user — these may differ from one user to another. To make things even more complicated, at some charging

parks the charging park operator will offer charging possibilities independent of any EM provider and bill these in other ways. For example, to name just a few, following billing models in such cases may occur (not an exhaustive list):

— *Based on usage (e.g. per hour, …)*

— *Combination with parking fee*

— *"Buying costumers" can charge for free or get a price reduction*

— *Dependent on maximum current ("slow" vs. "fast" charging)*

— *…*

TPEG EMI should support all the different, flexible, but amongst each other incompatible billing models. All the different options are very difficult to support in a single, fixed data structure. Moreover, doing so has a high risk that another, unforeseen billing model will be invented in the future by a provider, which cannot be covered by such a data structure.

Therefore EMI only refers to a "Price", to give end users an indication, e.g. of maximum or minimum price and to distinguish between normal and premium offers and to allow comparison of charging parks. Additional information, e.g. on special offers, can be given as free text.

6.4 EMI message structure

The structure of the top-level EMI message is illustrated in Figure 7. An EMI message is designed to either contain descriptive (largely static) information or availability (largely dynamic) information for a charging park and/or charging station.

The following types of information can be transmitted in an EMI message:

— Information on a charging park (typically: capacity, site description, etc.). Pricing/billing information can be included.

— Information on charging stations belonging to the charging park (typically: available connector types, supported vehicle types, vehicle size restrictions).

— Availability information for a charging park (number of free places).

— Availability information for a charging station (e.g. the number of free places per connector type).

— Location information on a charging park (in addition to the obligatory coordinates in the LocationReferencingContainer), optionally enhanced by location information for charging stations.

Typically, charging park information should be transmitted in an EMI message with a long expiry time as this type of information is very stable for a given service. EMI messages containing availability information can refer to this via a cross reference by the parkID_Key and the stationID_Key. Thus, EMI supports the transmission of full or partial descriptive information for charging parks and charging stations, which may be recombined in the TPEG receiver. This allows a service provider to fit the amount of information and repetition rate for EMI messages to the available data rate for his service.

The following is the minimum information needed for a TPEG client to present useful information to its user:

— ChargingParkInformation containing the parkID_Key and a minimal ChargingParkSiteDesription (parkName and the name of the parkOperator), ideally complemented by the

— ChargingParkAvailabilityVector containing a timestamp and minimal information on the ChargingParkAvailability (parkID_Key and overall number of freePlacesForPark).

The attributes of the ChargingParkInformation may be used by the end user to select an appropriate charging park and/or by the TPEG client to (automatically) filter out those charging parks not matching the end user's requirements.

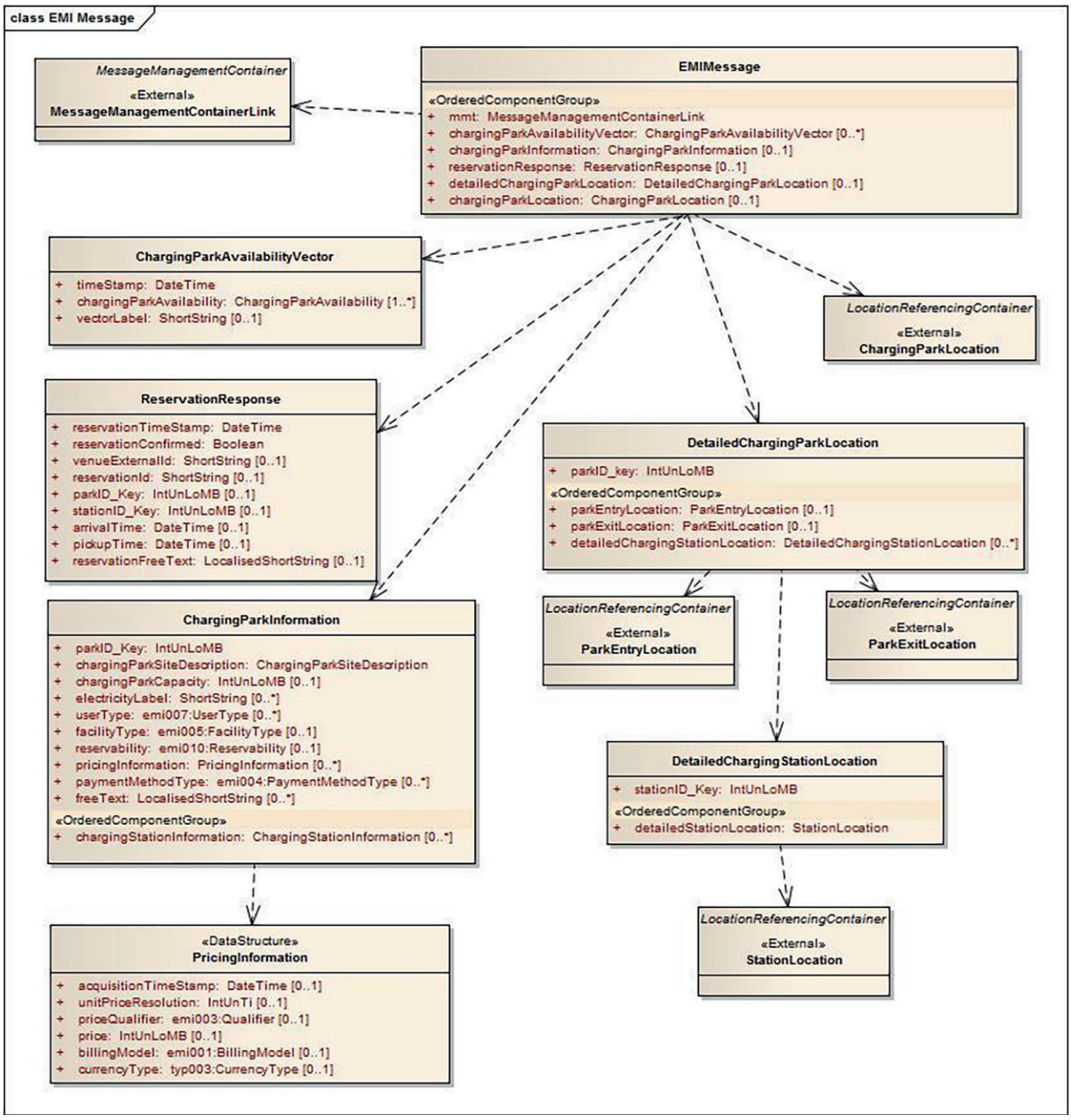

Figure 7 — Main structure of an EMI message

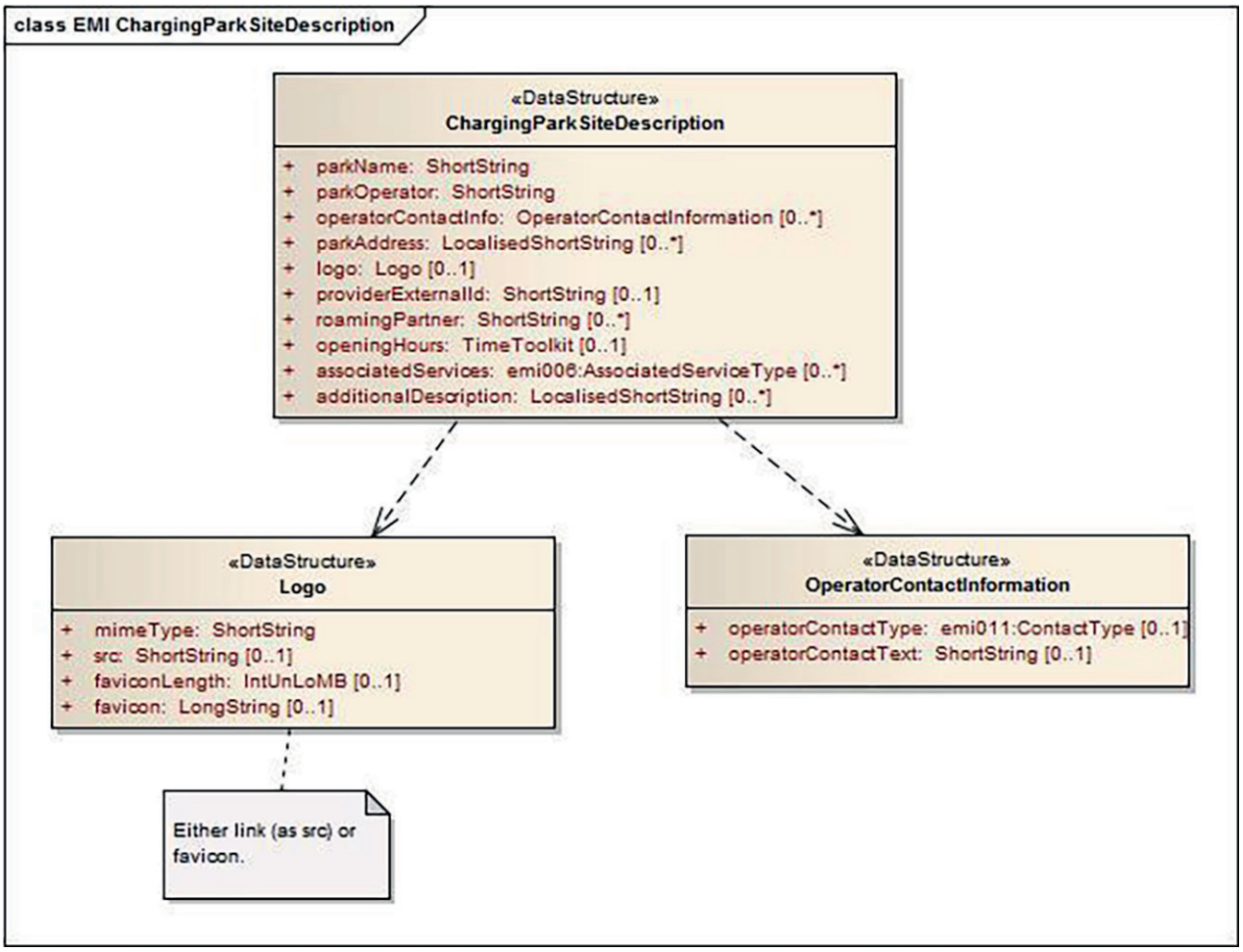

Figure 8 — Structure of EMI ChargingParkSiteDescription

The data structure ChargingParkSiteDescription extends the ChargingParkInformation with more detailed descriptive information on the ChargingPark; see Figure 8.

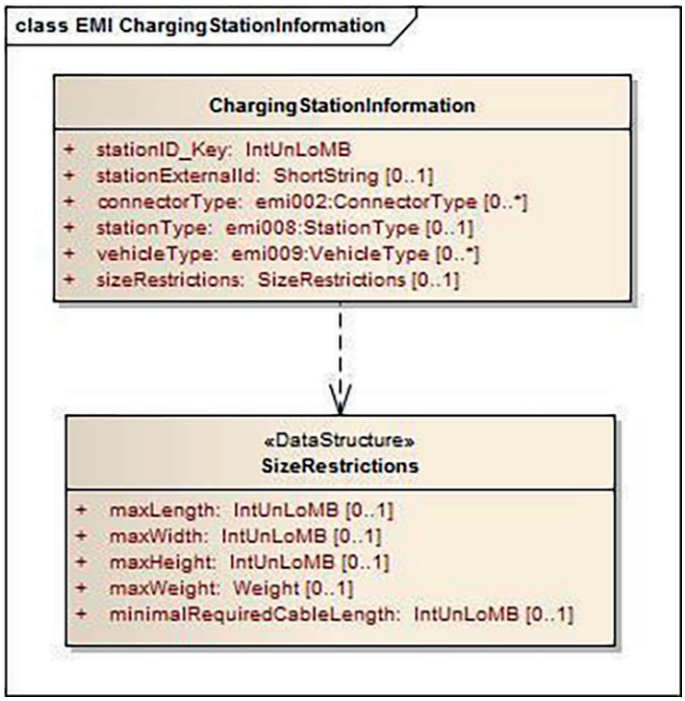

Figure 9 — Structure of EMI ChargingStationInformation

Information on charging stations can be added for each charging park; see Figure 9. Most importantly, this data structure allows the specification of the connector types which are supported by each charging station, allowing a filtering accordingly by the TPEG client to match this information with the socket available in the end user's vehicle. Similarly, also the other attributes may be used to filter out those charging stations not matching the end user's requirements and/or the electric vehicle's features.

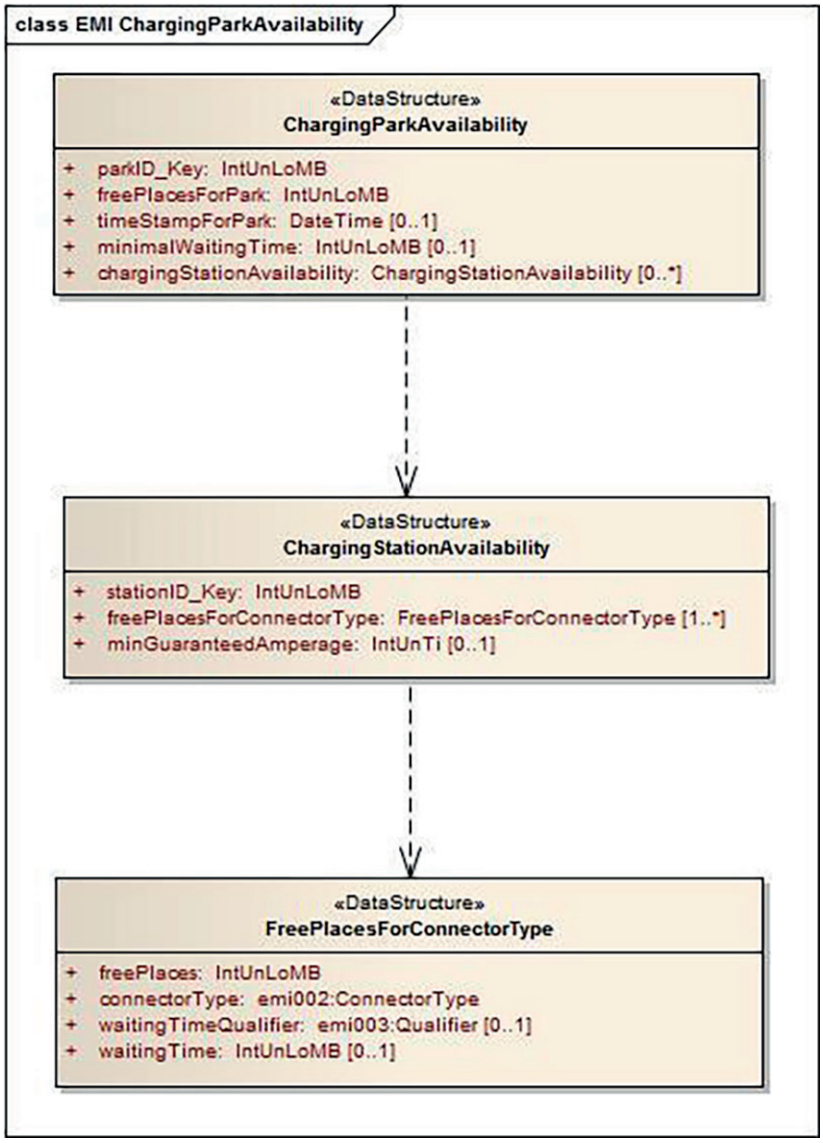

Figure 10 — Structure of EMI ChargingParkAvailability

A service provider can choose to provide availability information aggregated at the level of charging parks or to provide it at the level of charging stations; see Figure 10. When the service provider chooses to aggregate the availability information for each charging park, the total number of free charging places in this charging park shall be indicated.

In the case where the availability information is provided for charging stations, the service provider shall specify, for each charging station, how many charging places are available for each connector type (i.e. aggregated over the charging points at the respective charging station).

As each charging point at a charging station may contain several connectors of different types, the sum of all freePlacesForConnectorType (see Table 12) over all charging stations may be much higher than the freePlacesForPark (see Table 11). The service provider should take care to accurately calculate these figures aggregated at the different levels.

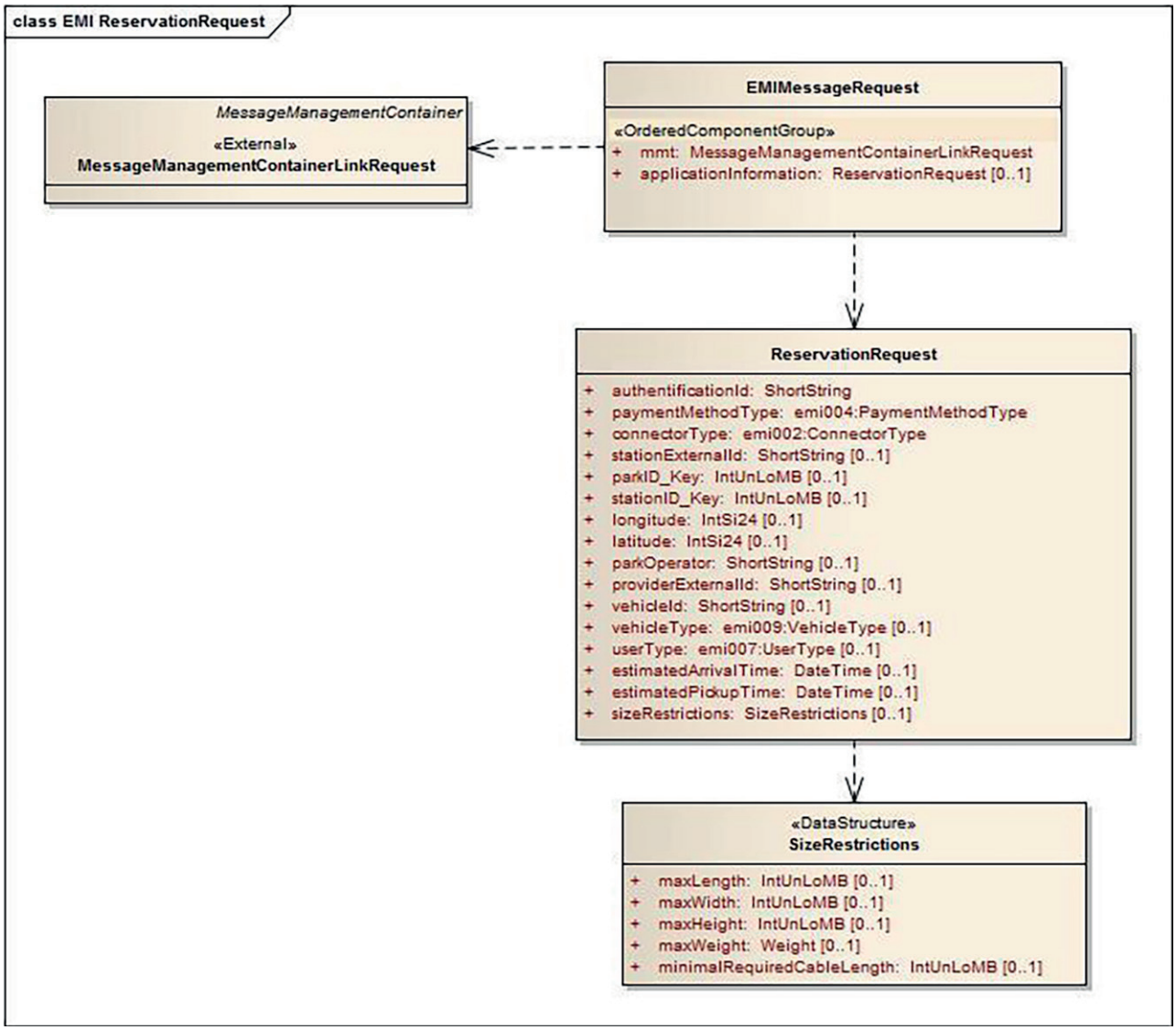

Figure 11 — Structure of EMI Request

The UML class structure of the EMI Request message is shown in Figure 11. The EMI response is contained in the main EMI message (see Figure 7).

In Annex A, the TPEG binary representation of this EMI service data frame structure is defined. In Annex B, the TPEG-ML representation for EMI is specified.

7 EMI message components

7.1 EMIMessage

The overall EMI message container is defined in Table 2.

Table 2 — EMIMessage

Name	Type	Multiplicity	Description
Ordered components			
mmt	MessageManagementContainer-Link	1	This component is specified in the MMC specification. It contains all and only information related to message management. The sender side, especially the instance generating the transmission data, has to ensure that the message management information allows unambiguous interpretation over time and in appropriate scenarios with disturbed reception specific to the transmission channel.
chargingParkAvailabilityVector	ChargingParkAvailabilityVector	0..*	This component groups information on the availability of charging stations in one or more charging parks. This can be highly dynamic information.
chargingParkInformation	ChargingParkInformation	0..1	Component to transmit static information on a charging park. This component can be sent with a low repetition rate.
reservationResponse	ReservationResponse	0..1	Component to transmit the response to a reservation request.
detailedChargingParkLocation	DetailedChargingParkLocation	0..1	Detailed geolocation referencing information for charging park entry, charging park exit or single charging stations.
chargingParkLocation	ChargingParkLocation	0..1	Geographic coverage location reference for a charging park. Not applicable for availability vector

7.2 MessageManagementContainerLink

The MessageManagementContainerLink is the MessageManagementContainer from the TPEG2-MMC toolkit in ISO/TS 21219-6.

7.3 ChargingParkLocation

The ChargingParkLocation is the Geographic Coverage location reference, Location Referencing Container from the TPEG-LRC toolkit in ISO/TS 18234-11.

7.4 ChargingParkAvailabilityVector

Table 3 shows the component that groups information on the availability of charging stations within one or more charging parks. This can be highly dynamic information.

Table 3 — ChargingParkAvailabilityVector

Name	Type	Multiplicity	Description
timeStamp	DateTime	1	Provides timing information for the latest valid information on the availability of charging places at a charging park
chargingParkAvailability	ChargingParkAvailability	1..*	Information on the availability of charging places at a specific charging park. Optionally, the availability information may be broken down to the level of charging stations.
vectorLabel	ShortString	0..1	Description of charging parks belonging to this vector (e.g. all charging parks in Garmisch-Partenkirchen)

7.5 ChargingParkInformation

Table 4 shows the component to transmit largely static information on a charging park. This component can be sent with a low repetition rate.

Table 4 — ChargingParkInformation

Name	Type	Multiplicity	Description
parkID_Key	IntUnLoMB	1	EMI internal identifier for a charging park. The parkID_Key shall be used by an end-user device or application to combine charging park information, detailed charging park location and ChargingParkAvailability data and, optionally, the ReservationResponse.
chargingParkSiteDescription	ChargingParkSiteDescription	1	Further background information on this charging park.
chargingParkCapacity	IntUnLoMB	0..1	Total number of available charging places at this charging park.
electricityLabel	ShortString	0..*	Identifier for the electricity provided at this charging park e.g. by which (renewable) energy source the electricity is generated.
userType	emi007:UserType	0..*	Information on types of users that are permitted to make use of this charging park (e.g. all users, customers, etc.).
facilityType	emi005:FacilityType	0..1	Type of charging park (e.g. private or public)
reservability	emi010:Reservability	0..1	Information as to whether reservation of a charging place is possible at this charging park.
pricingInformation	PricingInformation	0..*	Information on pricing and billing models at this charging park. EMI supports many different, flexible billing models.
paymentMethodType	emi004: PaymentMethodType	0..*	Payment methods supported at this charging park.
freeText	LocalizedShortString	0..*	Can contain additional information, e.g. regarding the billing options (such as promotional offers), at this charging park.
Ordered components			
chargingStationInformation	ChargingStationInformation	0..*	Static information on a charging station.

7.6 ChargingStationInformation

Table 5 shows the component to transmit static information on a charging station. This component may be sent with a low repetition rate.

Table 5 — ChargingStationInformation

Name	Type	Multiplicity	Description
stationID_Key	IntUnLoMB	1	EMI internal identifier for a charging station. Within a charging park (identified by its parkID_Key) the stationID_Key uniquely identifies a charging station. The stationID_Key shall be used by an end-user device or application to combine charging station information, detailed charging station location and ChargingStationAvailability data and, optionally, the ReservationResponse.
stationExternalId	ShortString	0..1	External identifier for the charging station, if applicable (e.g. based on the EVSE ID).
connectorType	emi002:ConnectorType	0..*	Information on the connector type(s) supported at this charging station.
stationType	emi008:StationType	0..1	Information on the station type for this charging station.
vehicleType	emi009:VehicleType	0..*	Information on the vehicle type(s) that are supported for using this charging station.
sizeRestrictions	SizeRestrictions	0..1	Applicable size restrictions on vehicles permitted to park and charge at this charging station.

7.7 DetailedChargingParkLocation

Table 6 shows the detailed geolocation referencing information for charging park entry, charging park exit or single charging stations.

Table 6 — DetailedChargingParkLocation

Name	Type	Multiplicity	Description
parkID_Key	IntUnLoMB	1	EMI internal identifier for a charging park. The parkID_Key shall be used by an end-user device or application to combine charging park information, detailed charging park location and ChargingParkAvailability data and, optionally, the ReservationResponse.
Ordered components			
parkEntryLocation	ParkEntryLocation	0..1	Additional information on the location of the entry roads of the charging park.
parkExitLocation	ParkExitLocation	0..1	Additional information on the location of the exit roads of the charging park.
detailedChargingStationLocation	DetailedChargingStationLocation	0..*	Container component holding detailed location information for a list of charging stations.

7.8 ParkEntryLocation

The ParkEntryLocation is the component to encode a charging park entrance road/ramp as a Location Referencing Container from the TPEG-LRC toolkit in ISO/TS 18234-11.

7.9 ParkExitLocation

The ParkExitLocation is the component to encode a charging park exit road/ramp as a Location Referencing Container from the TPEG-LRC toolkit in ISO/TS 18234-11.

7.10 DetailedChargingStationLocation

Table 7 shows the location reference information for the location of the charging station.

Table 7 — DetailedChargingStationLocation

Name	Type	Multiplicity	Description
stationID_Key	IntUnLoMB	1	EMI internal identifier for a charging station. Within a charging park (identified by its parkID_Key), the stationID_Key uniquely identifies a charging station. The stationID_Key shall be used by an end-user device or application to combine charging station information, detailed charging station location and ChargingStationAvailability data and, optionally, the the ReservationResponse.
Ordered components			
detailedStationLocation	StationLocation	1	Location reference for the charging station.

7.11 StationLocation

The StationLocation is the component to encode a charging station location as a Location Referencing Container from the TPEG-LRC toolkit in ISO/TS 18234-11.

7.12 EMIMessageRequest

Table 8 shows the overall EMIMessageRequest container.

Table 8 — EMIMessageRequest

Name	Type	Multiplicity	Description
Ordered components			
mmt	MessageManagementContainerLinkRequest	1	This component is specified in the MMC specification. It contains all and only information related to message management. The sender side, especially the instance generating the transmission data, has to ensure that the message management information allows unambiguous interpretation over time and in appropriate scenarios with disturbed reception specific to the transmission channel.
applicationInformation	ReservationRequest	0..1	Component to transmit a reservation request.

7.13 MessageManagementContainerLinkRequest

The MessageManagementContainerLinkRequest is the MessageManagementContainer from the TPEG-MMC toolkit in ISO/TS 21219-6.

7.14 ReservationRequest

Table 9 defines the component to transmit a ReservationRequest.

Table 9 — ReservationRequest

Name	Type	Multiplicity	Description
authentificationId	ShortString	1	An ID that authenticates the user that initiates the reservation request. The authentication can be handled by means of the customer card issued by the EM provider.
paymentMethodType	emi004: PaymentMethodType	1	Payment method(s) that the requesting user can support. These should be matched against the payment methods supported by the charging park. This information is used to ensure the billing of the reservation.
connectorType	emi002:ConnectorType	1	Information on the connector type(s) supported by the electric vehicle for which the reservation request is carried out.
stationExternalId	ShortString	0..1	External identifier for the charging station, if known (e.g. based on the EVSE ID), for which a reservation is requested. No further geographical identification information is needed for the request.

Table 9 *(continued)*

Name	Type	Multiplicity	Description
parkID_Key	IntUnLoMB	0..1	EMI internal identifier for a charging park for which a reservation is requested. Typically, the Reservation Request will also use the stationID_Key to identify the charging station within the charging park. The parkID_Key shall be used by an end-user device or application to combine charging park information, detailed charging park location and ChargingParkAvailability data and, optionally, the ReservationResponse.
stationID_Key	IntUnLoMB	0..1	EMI internal identifier for a charging station for which a reservation is requested. When this identifier is used in a request, the parkID_Key shall also be specified. Within a charging park (identified by its parkID_Key), the stationID_Key uniquely identifies a charging station. The stationID_Key shall be used by an end-user device or application to combine charging station information, detailed charging station location and ChargingStationAvailability data and, optionally, the ReservationResponse.
longitude	IntSi24	0..1	Longitude of a location at which a reservation is requested. Shall only be used in a request in combination with latitude. Together, the longitude and latitude coordinates identify a location at or near which a charging point should be reserved.
latitude	IntSi24	0..1	Latitude of a location at which a reservation is requested. Shall only be used in a request in combination with longitude. Together, the longitude and latitude coordinates identify a location at or near which a charging point should be reserved.
parkOperator	ShortString	0..1	Identification of the charging park operator. May be used in combination with longitude/latitude, in the case where the Reservation Request is done for a charging point operated by a specific operator.
providerExternalId	ShortString	0..1	External Identifier for the electromobility provider, if applicable (e.g. based on the EVSE ID).
vehicleId	ShortString	0..1	Identifier for the vehicle, if applicable (e.g. based on the car number plate).
vehicleType	emi009:VehicleType	0..1	Information on the Vehicle Type for which the Reservation Request is performed.
userType	emi007:UserType	0..1	Information on type(s) of user that is carrying out the Reservation Request.

Table 9 *(continued)*

Name	Type	Multiplicity	Description
estimatedArrivalTime	DateTime	0..1	Estimated time of arrival at the location at which a reservation is requested.
estimatedPickupTime	DateTime	0..1	Estimated time at which the vehicle will be retrieved from the location at which a reservation is requested.
sizeRestrictions	SizeRestrictions	0..1	Dimensions of the vehicle for which a reservation is requested.

7.15 ReservationResponse

Table 10 shows the component to transmit the response to a reservation request. The component shall contain all relevant information on a charging station for which a reservation request can be confirmed.

Table 10 — ReservationResponse

Name	Type	Multiplicity	Description
reservationTimeStamp	DateTime	1	Time at which the reservation response is given.
reservationConfirmed	Boolean	1	True, if reservation is confirmed; otherwise, false.
venueExternalId	ShortString	0..1	External identifier for charging point or charging station (e.g. based on the EVSE ID), for which the reservation response is given. Usage of this attribute is recommended in case of a positive reservation response.
reservationId	ShortString	0..1	Unique identifier for the reservation response. Usage of this attribute is recommended in the case of a positive reservation response.
parkID_Key	IntUnLoMB	0..1	EMI internal identifier for the charging park for which the reservation is confirmed. Typically, the reservation response will also use the stationID_Key to identify the charging station within the charging park. The parkID_Key shall be used by an end-user device or application to combine charging park information, detailed charging park location and ChargingParkAvailability data and, optionally, the ReservationResponse.
stationID_Key	IntUnLoMB	0..1	EMI internal identifier for the charging station for which the reservation is confirmed. When this identifier is used, also the parkID_Key shall be specified. Within a charging park (identified by its parkID_Key), the stationID_Key uniquely identifies a charging station. The stationID_Key shall be used by an end-user device or application to combine charging station information, detailed charging station location and ChargingStationAvailability data and, optionally, the ReservationResponse.

Table 10 *(continued)*

Name	Type	Multiplicity	Description
arrivalTime	DateTime	0..1	Earliest time at which the charging point, for which the reservation is confirmed, will be available.
pickupTime	DateTime	0..1	Latest time at which the vehicle shall be removed from the charging point for which the reservation is confirmed.
reservationFreeText	LocalizedShortString	0..1	Any additional information regarding the reservation which is confirmed, e.g. on cancellation upon non-arrival, billing of the time the charging point is kept available but not currently occupied, etc.

8 EMI Datatypes

8.1 ChargingParkAvailability

Table 11 shows the component to transmit information on the availability of charging places at a specific charging park. This may be highly dynamic information. Optionally, the availability information may be broken down to the level of charging stations and charging points.

Table 11 — ChargingParkAvailability

Name	Type	Multiplicity	Description
parkID_Key	IntUnLoMB	1	EMI internal identifier for a charging park. The parkID_Key shall be used by an end-user device or application to combine charging park information, detailed charging park location and ChargingParkAvailability data and, optionally, the ReservationResponse.
freePlacesForPark	IntUnLoMB	1	Current information on the overall number of free charging places within this charging park. This may be highly dynamic information. Note: This number does not have to be the sum of freePlacesForConnectorType data (see subcomponent ChargingStationAvailability) because a single charging station may have multiple connectors per parking spot.
timeStampForPark	DateTime	0..1	Provides timestamp giving date, hour, minute at which the latest valid information on ChargingParkAvailability was provided.
minimalWaitingTime	IntUnLoMB	0..1	Minimal estimated time in minutes an end user will have to wait if there are no available charging places in this charging park.
chargingStationAvailability	ChargingStationAvailability	0..*	For each charging station of this charging park optionally specifies the availability information. A charging park may contain multiple charging stations.

8.2 ChargingStationAvailability

Table 12 shows the component to transmit information on the availability of a charging station at a specific charging park.

Table 12 — ChargingStationAvailability

Name	Type	Multiplicity	Description
stationID_Key	IntUnLoMB	1	EMI internal identifier for a charging station. Within a charging park (identified by its parkID_Key), the stationID_Key uniquely identifies a charging station. The stationID_Key shall be used by an end-user device or application to combine charging station information, detailed charging station location and ChargingStationAvailability data and, optionally, the ReservationResponse.
freePlacesForConnectorType	FreePlacesForConnectorType	1..*	Current information giving the number of free charging places for each connector type at this charging station. This may be highly dynamic information.
minGuaranteedAmperage	IntUnTi	0..1	The minimal amount of current in Ampere which is currently guaranteed by the charging park operator. This information can be used by an end user or end-user device to estimate charging time.

8.3 FreePlacesForConnectorType

Table 13 shows the component to transmit dynamic information on a charging station specifying the number of available charging places per connector type at this charging station.

Table 13 — FreePlacesForConnectorType

Name	Type	Multiplicity	Description
freePlaces	IntUnLoMB	1	Current information giving the number of free charging places for this connector type.
connectorType	emi002:ConnectorType	1	Connector type.
waitingTimeQualifier	emi003:Qualifier	0..1	For usage with waiting time only. Qualifies the given waiting time.
waitingTime	IntUnLoMB	0..1	Estimated time in minutes an end user will have to wait if there are no available charging places at this charging station. Can be quantified by means of the waitingTimeQuantifier.

8.4 ChargingParkSiteDescription

Table 14 shows the component to transmit further background information on a charging park.

Table 14 — ChargingParkSiteDescription

Name	Type	Multiplicity	Description
parkName	ShortString	1	Name of individual charging park.
parkOperator	ShortString	1	Name of operator of charging park.
operatorContactInfo	OperatorContactInformation	0..*	Contact information for the operator of the charging park.
parkAddress	LocalizedShortString	0..*	Street address of the charging park in the relevant local language(s).
logo	Logo	0..1	Logo of the operator of the charging park.
providerExternalId	ShortString	0..1	External Identifier for the electromobility provider, if applicable (e.g. based on the EVSE ID).
roamingPartner	ShortString	0..*	Information on electromobility providers which have roaming agreements with the operator of this charging park. Based on this information, an end user can verify if charging at this charging park is interesting from a financial point of view. In the case, where the electromobility provider which the user has a contract with is not in the list of roamingPartner, use of this charging park might not be attractive.
openingHours	TimeToolkit	0..1	Opening hours of the charging park.
associatedServices	emi006:AssociatedServiceType	0..*	Lists available services which the charging park offers in addition to a charging facility (e.g. park-and-ride, access to public transportation, etc.).
additionalDescription	LocalizedShortString	0..*	Additional descriptive information on this charging park.

8.5 Logo

Table 15 shows the component to transmit operator Logo either as link or as favicon.

Table 15 — Logo

Name	Type	Multiplicity	Description
mimeType	ShortString	1	Mime type
src	ShortString	0..1	Link (URL) to the logo
faviconLength	IntUnLoMB	0..1	Length of favicon, max 1 024 bytes
favicon	LongString	0..1	Contents of favicon, N bytes, max 1 024 bytes

8.6 OperatorContactInformation

Table 16 shows the general information (e.g. phone, fax, email, website) to contact the operator of this charging park.

Table 16 — OperatorContactInformation

Name	Type	Multiplicity	Description
operatorContactType	emi011:ContactType	0..1	Contact type
operatorContactText	ShortString	0..1	Text containing phone number, etc.

8.7 SizeRestrictions

Table 17 shows the applicable size restrictions on vehicles allowed to park and charge at this charging station.

Table 17 — SizeRestrictions

Name	Type	Multiplicity	Description
maxLength	IntUnLoMB	0..1	Maximum length of the vehicle in centimetres.
maxWidth	IntUnLoMB	0..1	Maximum width of the vehicle in centimetres.
maxHeight	IntUnLoMB	0..1	Maximum height of the vehicle in centimetres.
maxWeight	Weight	0..1	Maximum weight of the vehicle in kilogrammes.
minimalRequiredCa-bleLength	IntUnLoMB	0..1	Minimal required length of charging cable in centimetres.

8.8 PricingInformation

Table 18 shows the information on pricing and billing models at this charging park. EMI supports many different, flexible billing models.

Table 18 — PricingInformation

Name	Type	Multiplicity	Description
acquisitionTimeStamp	DateTime	0..1	Time at which the last valid pricing information was published.
unitPriceResolution	IntUnTi	0..1	Resolution of unit price (decimal points).
priceQualifier	emi003:Qualifier	0..1	Qualification of the price information.
price	IntUnLoMB	0..1	Price per delivery unit. May be qualified by means of the qualifier.
billingModel	emi001:BillingModel	0..1	The unit for which unit prices are given.
currencyType	typ003:CurrencyType	0..1	Supported currency type(s) for payment at this charging park.

9 EMI Tables

9.1 emi001:BillingModel

Table 19 enumerates the types of billing models offered at this facility.

Table 19 — BillingModel

Code	Reference-English “word”	Comment	Example
000	unknown		
001	price per KWh	kilowatt-hour; unit, KWh	
002	price per hour		
003	price per minute		
004	flat rate		
005	free		
006	according to contract		
255	undefined		

9.2 emi002:ConnectorType

Table 20 enumerates the list of connector types supported at this facility, including (wherever appropriate) the technical maximum for supported charging current.

Table 20 — ConnectorType

Code	Reference-English "word"	Comment	Example
000	unknown		
001	Typ2 (EU IEC 62196–2), 400 V, 16 A		
002	Typ2 (EU IEC 62196–2), 400 V, 32 A		
003	Typ2 (EU IEC 62196–2), 400 V, 63 A		
004	CEE5, 400 V, 16 A		
005	CEE5, 400 V, 32 A		
006	CEE5, 400 V, 63 A		
007	CEE3, 230 V, 16 A		
008	Schuko, 230 V, 16 A		
009	Yazaki, 400 VDC, 125 A		
010	Chademo, 600 VDC		
011	BS 1363		
012	MK Commando K9785, 200 V to 250 V, 32 A		
013	Tesla connector		
014	NEMA 5, 125 V, 20 A		
015	NEMA 14–30, 250 V, 30 A		
016	NEMA 14–50, 250 V, 50 A		
017	NEMA 6–20, 240 V, 20 A		
018	SEV Type 13		
019	SEV Type 15		
020	SEV Type 23		
021	SEV Type 25		
022	Type E		
023	Type G		
024	Type I		
025	EN 50075 (Europlug)		
026	CCS (EU IEC 62196), DC + AC		
027	SAE J1772 AC Level1		
028	SAE J1772 AC Level2		
029	SAE J1772 AC Level3		
030	SAE J1772 DC Level1		
031	SAE J1772 DC Level2		
032	SAE J1772 DC Level3		
255	undefined		

9.3 emi003:Qualifier

Table 21 enumerates a list of qualifiers.

Table 21 — Qualifier

Code	Reference-English "word"	Comment	Example
000	unknown		
001	maximum		
002	minimum		
003	approximately		
004	exactly		
255	undefined		

9.4 emi004:PaymentMethodType

Table 22 enumerates the applicable methods of payment accepted at this facility.

Table 22 — PaymentMethodType

Code	Reference-English "word"	Comment	Example
000	unknown		
001	cash		
002	credit card		
003	bank card		
004	tank card		
006	debit card		
007	RFID		
008	prepaid card		
009	mobile phone		
010	smartcard		
011	customer loyalty card	In the UK, also known as "agency card"; in North America as "reward card" or "club card".	
255	undefined		

9.5 emi005:FacilityType

Table 23 enumerates a list of facility types.

Table 23 — FacilityType

Code	Reference-English "word"	Comment	Example
000	unknown		
001	public (on the street)		
002	public (parking)		
003	private		
004	semi-public		
005	commercial		
255	undefined		

9.6 emi006:AssociatedServiceType

Table 24 enumerates a list of applicable associated services which may be offered by a charging park.

Table 24 — AssociatedServiceType

Code	Reference-English "word"	Comment	Example
000	unknown		
001	public toilets		
002	mini-market		
003	carwash		
004	restaurant		
005	hotel		
006	vehicle repair shop		
007	internet hotspot		
008	parking		
009	park-and-drive		
010	park-and-ride		
011	airport		
012	railway station		
013	tourist centre		
255	undefined		

9.7 emi007:UserType

Table 25 enumerates a list of user types.

Table 25 — UserType

Code	Reference-English "word"	Comment	Example
000	unknown		
001	all users		
002	provider customers		
003	provider and roaming customers		
004	shoppers		
005	hotel guests		
006	staff		
007	employees		
008	visitors		
009	members		
255	undefined		

9.8 emi008:StationType

Table 26 enumerates a list of station types.

Table 26 — StationType

Code	Reference-English "word"	Comment	Example
000	unknown		
001	splash water-proof		
002	covered		
003	charge-and-park		
255	undefined		

9.9 emi009:VehicleType

Table 27 enumerates a list of vehicle types.

Table 27 — VehicleType

Code	Reference-English "word"	Comment	Example
000	unknown		
001	car		
002	bicycle		
003	tricycle		
004	scooter		
005	bus		
006	boat		
007	paddleboat		
008	personal transporter		
009	airplane		
255	undefined		

9.10 emi010:Reservability

Table 28 enumerates a list of reservability types.

Table 28 — Reservability

Code	Reference-English "word"	Comment	Example
000	unknown		
001	reservation required		
002	reservation recommended		
003	not reservable		
255	undefined		

9.11 emi011:ContactType

Table 29 enumerates a list of contact types.

Table 29 — ContactType

Code	Reference-English "word"	Comment	Example
000	unknown		
001	telephone		
002	fax		
003	email		
004	internet address		
005	priority telephone		
006	main office telephone		
255	undefined		

Annex A
(normative)

TPEG application, TPEG-Binary Representation

A.1 Message components

A.1.1 List of generic component IDs

Table A.1 — Generic component IDs

Name	ID
EMIMessage	0
MessageManagementContainerLink	1
ChargingParkLocation	4
ChargingParkAvailabilityVector	5
ChargingParkInformation	6
ChargingStationInformation	7
DetailedChargingParkLocation	8
ParkEntryLocation	9
ParkExitLocation	10
DetailedChargingStationLocation	11
StationLocation	12
EMIMessageRequest	14
MessageManagementContainerLinkRequest	15
ReservationRequest	16
ReservationResponse	17

A.1.2 EMIMessage

<EMIMessage(0)>:=	
<IntUnTi>(0),	: ID of this component
<IntUnLoMB>(lengthComp),	: Number of bytes in this component
<IntUnLoMB>(lengthAttr),	: Number of bytes in attributes
ordered {	

Table A.1 *(continued)*

<MessageManagementContainerLink>(mmt),	: This component is specified in the MMC specification. It contains all and only information related to message management. The sender side, especially the instance generating the transmission data, has to ensure that the message management information allows unambiguous interpretation over time and in appropriate scenarios with disturbed reception specific to the transmission channel.
n *<ChargingParkAvailabilityVector>(chargingParkAvailabilityVector),	: This component groups information on the availability of charging stations in one or more charging parks. This may be highly dynamic information.
n *<ChargingParkInformation>(chargingParkInformation)[0..1],	: Component to transmit static information on a charging park. This component may be sent with a low repetition rate.
n *<ReservationResponse>(reservationResponse)[0..1],	: Component to transmit the response to a reservation request.
n *<DetailedChargingParkLocation>(detailedChargingParkLocation)[0..1],	: Detailed geolocation referencing information for charging park entry, charging park exit or single charging stations.
n *<ChargingParkLocation>(chargingParkLocation)[0..1]	: Geographic coverage location reference for a charging park. Not applicable for availability vector.
};	

A.1.3 MessageManagementContainerLink

<MessageManagementContainerLink(1)>:=	
External<MessageManagementContainer(1)>;	: See MessageManagementContainer specification.

A.1.4 ChargingParkLocation

<ChargingParkLocation(4)>:=	
External<LocationReferencingContainer(4)>;	: See LocationReferencingContainer specification.

A.1.5 ChargingParkAvailabilityVector

<ChargingParkAvailabilityVector(5)>:=	
<IntUnTi>(5),	: ID of this component
<IntUnLoMB>(lengthComp),	: Number of bytes in this component
<IntUnLoMB>(lengthAttr),	: Number of bytes in attributes
<DateTime>(timeStamp),	: Provides timing information for the latest valid information on the availability of charging places at a charging park.

<IntUnLoMB>(n),	
n *<ChargingParkAvailability>(chargingParkAvailability),	: Information on the availability of charging places at a specific charging park. Optionally, the availability information may be broken down to the level of charging stations.
BitArray(selector),	
if (bit 0 of selector is set)	
<ShortString>(vectorLabel);	: Description of charging parks belonging to this vector (e.g. all charging parks in Garmisch-Partenkirchen).

A.1.6 ChargingParkInformation

<ChargingParkInformation(6)>:=	
<IntUnTi>(6),	: ID of this component
<IntUnLoMB>(lengthComp),	: Number of bytes in this component
<IntUnLoMB>(lengthAttr),	: Number of bytes in attributes
<IntUnLoMB>(parkID_Key),	: EMI internal identifier for a charging park. The parkID_Key shall be used by an end-user device or application to combine charging park information, detailed charging park location and ChargingParkAvailability data and, optionally, the ReservationResponse.
<ChargingParkSiteDescription>(chargingParkSiteDescription),	: Further background information on this charging park.
BitArray(selector),	
if (bit 0 of selector is set)	
<IntUnLoMB>(chargingParkCapacity),	: Total number of available charging places at this charging park.
if (bit 1 of selector is set)	
{	
<IntUnLoMB>(n),	
n *<ShortString>(electricityLabel),	: Identifier for the electricity provided at this charging park, by which (renewable) energy source the electricity is generated.
}	
if (bit 2 of selector is set)	
{	
<IntUnLoMB>(n),	
n *<emi007:UserType>(userType),	: Information on types of users that are permitted to make use of this charging park (e.g. all users, customers, etc.).
}	
if (bit 3 of selector is set)	
<emi005:FacilityType>(facilityType),	: Type of charging park (e.g. private or public).

if (bit 4 of selector is set)	
<emi010:Reservability>(reservability),	: Information as to whether reservation of a charging place is possible at this charging park.
if (bit 5 of selector is set)	
{	
<IntUnLoMB>(n),	
n *<PricingInformation>(pricingInformation),	: Information on pricing and billing models at this charging park. EMI supports many different flexible billing models.
}	
if (bit 6 of selector is set)	
{	
<IntUnLoMB>(n),	
n *<emi004:PaymentMethodType>(paymentMethodType),	: Payment methods supported at this charging park.
}	
if (bit 7 of selector is set)	
{	
<IntUnLoMB>(n),	
n *<LocalizedShortString>(freeText),	: May contain additional information regarding the billing options (such as promotional offers) at this charging park.
}	
ordered {	
n *<ChargingStationInformation>(chargingStationInformation)	: Static information on a charging station.
};	

A.1.7 ChargingStationInformation

<ChargingStationInformation(7)>:=	
<IntUnTi>(7),	: ID of this component
<IntUnLoMB>(lengthComp),	: Number of bytes in this component
<IntUnLoMB>(lengthAttr),	: Number of bytes in attributes
<IntUnLoMB>(stationID_Key),	: EMI internal identifier for a charging station. Within a charging park (identified by its parkID_Key), the stationID_Key uniquely identifies a charging station. The stationID_Key shall be used by an end-user device or application to combine charging station information, detailed charging station location and ChargingStationAvailability data and, optionally, the ReservationResponse.

BitArray(selector),	
if (bit 0 of selector is set)	
<ShortString>(stationExternalId),	: External identifier for the charging station, if applicable (e.g. based on the EVSE ID).
if (bit 1 of selector is set)	
{	
<IntUnLoMB>(n),	
n *<emi002:ConnectorType>(connectorType),	: Information on the connector type(s) supported at this charging station.
}	
if (bit 2 of selector is set)	
<emi008:StationType>(stationType),	: Information on the station type for this charging station.
if (bit 3 of selector is set)	
{	
<IntUnLoMB>(n),	
n *<emi009:VehicleType>(vehicleType),	: Information on the vehicle type(s) that are supported for using this charging station.
}	
if (bit 4 of selector is set)	
<SizeRestrictions>(sizeRestrictions);	: Applicable size restrictions on vehicles permitted to park and charge at this charging station.

A.1.8 DetailedChargingParkLocation

<DetailedChargingParkLocation(8)>:=	
<IntUnTi>(8),	: ID of this component
<IntUnLoMB>(lengthComp),	: Number of bytes in this component
<IntUnLoMB>(lengthAttr),	: Number of bytes in attributes
<IntUnLoMB>(parkID_Key),	: EMI internal identifier for a charging park. The parkID_Key shall be used by an end-user device or application to combine charging park information, detailed charging park location and ChargingParkAvailability data and, optionally, the ReservationResponse.
ordered {	
n *<ParkEntryLocation>(parkEntryLocation)[0..1],	: Additional information on the location of the entry roads of the charging park.

n *<ParkExitLocation>(parkExitLocation)[0..1],	: Additional information on the location of the exit roads of the charging park.
n *<DetailedChargingStationLocation>(detailedChargingStationLocation)	: Container component holding detailed location information for a list of charging stations.
};	

A.1.9 ParkEntryLocation

<ParkEntryLocation(9)>:=	
External <LocationReferencingContainer(9)>;	: See LocationReferencingContainer specification.

A.1.10 ParkExitLocation

<ParkExitLocation(10)>:=	
External <LocationReferencingContainer(10)>;	: See LocationReferencingContainer specification.

A.1.11 DetailedChargingStationLocation

<DetailedChargingStationLocation(11)>:=	
<IntUnTi>(11),	: ID of this component
<IntUnLoMB>(lengthComp),	: Number of bytes in this component
<IntUnLoMB>(lengthAttr),	: Number of bytes in attributes
<IntUnLoMB>(stationID_Key),	: EMI internal identifier for a charging station. Within a charging park (identified by its parkID_Key), the stationID_Key uniquely identifies a charging station. The stationID_Key shall be used by an end-user device or application to combine charging station information, detailed charging station location and ChargingStationAvailability data and, optionally, the ReservationResponse.
ordered {	
<StationLocation>(detailedStationLocation)	: Location reference for the charging station.
};	

A.1.12 StationLocation

<StationLocation(12)>:=	
External <LocationReferencingContainer(12)>;	: See LocationReferencingContainer specification.

A.1.13 EMIMessageRequest

<EMIMessageRequest(14)>:=	
<IntUnTi>(14),	: ID of this component
<IntUnLoMB>(lengthComp),	: Number of bytes in this component

<IntUnLoMB>(lengthAttr),	: Number of bytes in attributes
ordered {	
<MessageManagementContainerLinkRequest>(mmt),	: This component is specified in the MMC specification. It contains all and only information related to message management. The sender side, especially the instance generating the transmission data has to ensure that the message management information allows unambiguous interpretation over time and in appropriate scenarios with disturbed reception specific to the transmission channel.
n *<ReservationRequest>(applicationInformation)[0..1]	: Component to transmit a reservation request.
};	

A.1.14 MessageManagementContainerLinkRequest

<MessageManagementContainerLink-Request(15)>:=	
External<MessageManagementContainer(15)>;	: See MessageManagementContainer specification.

A.1.15 ReservationRequest

<ReservationRequest(16)>:=	
<IntUnTi>(16),	: ID of this component
<IntUnLoMB>(lengthComp),	: Number of bytes in this component
<IntUnLoMB>(lengthAttr),	: Number of bytes in attributes
<ShortString>(authentificationId),	: An ID that authenticates the user that initiates the reservation request. The authentication can be handled by means of the customer card issued by the EM provider.
<emi004:PaymentMethodType>(paymentMethodType),	: Payment method(s) that the requesting user can support. These should be matched against the payment methods supported by the charging park. This information is used to ensure the billing of the reservation.
<emi002:ConnectorType>(connectorType),	: Information on the connector type(s) supported by the electric vehicle for which the reservation request is carried out.
BitArray(selector),	
if (bit 0 of selector is set)	

<ShortString>(stationExternalId),	: External identifier for the charging station, if known (e.g. based on the EVSE ID), for which a reservation is requested. No further geographical identification information is needed for the request.
if (bit 1 of selector is set)	
<IntUnLoMB>(parkID_Key),	: EMI internal identifier for a charging park for which a reservation is requested. Typically, the Reservation Request will also use the stationID_Key to identify the charging station within the charging park. The parkID_Key shall be used by an end-user device or application to combine charging park information, detailed charging park location and ChargingParkAvailability data and, optionally, the ReservationResponse.
if (bit 2 of selector is set)	
<IntUnLoMB>(stationID_Key),	: EMI internal identifier for a charging station for which a reservation is requested. When this identifier is used in a request, also the parkID_Key shall be specified. Within a charging park (identified by its parkID_Key), the stationID_Key uniquely identifies a charging station. The stationID_Key shall be used by an end-user device or application to combine charging station information, detailed charging station location and ChargingStationAvailability data and, optionally, the ReservationResponse.
if (bit 3 of selector is set)	
<IntSi24>(longitude),	: Longitude of a location at which a reservation is requested. Shall only be used in a request in combination with latitude. Together, the longitude and latitude coordinates identify a location at or near which a charging point should be reserved.
if (bit 4 of selector is set)	
<IntSi24>(latitude),	: Latitude of a location at which a reservation is requested. Shall only be used in a request in combination with longitude. Together, the longitude and latitude coordinates identify a location at or near which a charging point should be reserved.
if (bit 5 of selector is set)	

<ShortString>(parkOperator),	: Identification of the charging park operator. May be used in combination with longitude/latitude, in the case where the Reservation Request is done for a charging point operated by a specific operator.
if (bit 6 of selector is set)	
<ShortString>(providerExternalId),	: External identifier for the electromobility provider, if applicable (e.g. based on the EVSE ID).
if (bit 7 of selector is set)	
<ShortString>(vehicleId),	: Identifier for the vehicle, if applicable (e.g. based on the car number plate).
if (bit 8 of selector is set)	
<emi009:VehicleType>(vehicleType),	: Information on the vehicle type for which the reservation request is performed.
if (bit 9 of selector is set)	
<emi007:UserType>(userType),	: Information on type(s) of user that is carrying out the reservation request.
if (bit 10 of selector is set)	
<DateTime>(estimatedArrivalTime),	: Estimated time of arrival at the location at which a reservation is requested.
if (bit 11 of selector is set)	
<DateTime>(estimatedPickupTime),	: Estimated time at which the vehicle will be retrieved from the location at which a reservation is requested.
if (bit 12 of selector is set)	
<SizeRestrictions>(sizeRestrictions);	: Dimensions of the vehicle for which a reservation is requested.

A.1.16 ReservationResponse

<ReservationResponse(17)>:=	
<IntUnTi>(17),	: ID of this component.
<IntUnLoMB>(lengthComp),	: Number of bytes in this component
<IntUnLoMB>(lengthAttr),	: Number of bytes in attributes
<DateTime>(reservationTimeStamp),	: Time at which the reservation response is given.
BitArray(selector),	
if (bit 0 of selector is set)	
<Boolean>(reservationConfirmed),	: True, if reservation is confirmed; otherwise, false.
if (bit 1 of selector is set)	

<ShortString>(venueExternalId),	: External Identifier for charging point or charging station (e.g. based on the EVSE ID), for which the reservation response is given. Usage of this attribute is recommended in case of a positive reservation response.
if (bit 2 of selector is set)	
<ShortString>(reservationId),	: Unique identifier for the reservation response. Usage of this attribute is recommended in the case of a positive reservation response.
if (bit 3 of selector is set)	
<IntUnLoMB>(parkID_Key),	: EMI internal identifier for the charging park for which the reservation is confirmed. Typically, the Reservation Response will also use the stationID_Key to identify the charging station within the charging park. The parkID_Key shall be used by an end-user device or application to combine charging park information, detailed charging park location and ChargingParkAvailability data and, optionally, the ReservationResponse.
if (bit 4 of selector is set)	
<IntUnLoMB>(stationID_Key),	: EMI internal identifier for the charging station for which the reservation is confirmed. When this identifier is used, also the parkID_Key shall be specified. Within a charging park (identified by its parkID_Key), the stationID_Key uniquely identifies a charging station. The stationID_Key shall be used by an end-user device or application to combine charging station information, detailed charging station location and ChargingStationAvailability data and, optionally, the ReservationResponse.
if (bit 5 of selector is set)	
<DateTime>(arrivalTime),	: Earliest time at which the charging point for which the reservation is confirmed, will be available.
if (bit 6 of selector is set)	

<DateTime>(pickupTime),	: Latest time at which the vehicle shall be removed from the charging point for which the reservation is confirmed.
if (bit 7 of selector is set)	
<LocalizedShortString>(reservationFreeText);	: Any additional information regarding the reservation which is confirmed, e.g. on cancellation upon non-arrival, billing of the time the charging point is kept available but not currently occupied, etc.

A.2 EMI Datatypes

A.2.1 ChargingParkAvailability

<ChargingParkAvailability>:=	
<IntUnLoMB>(parkID_Key),	: EMI internal identifier for a charging park. The parkID_Key shall be used by an end-user device or application to combine charging park information, detailed charging park location and ChargingParkAvailability data and, optionally, the ReservationResponse.
<IntUnLoMB>(freePlacesForPark),	: Current information on the overall number of free charging places within this charging park. This may be highly dynamic information. NOTE This number does not have to be the sum of freePlacesForConnectorType (see subcomponent ChargingStationAvailability) because a single charging station may have multiple connectors per parking spot.
BitArray(selector),	
if (bit 0 of selector is set)	
<DateTime>(timeStampForPark),	: Provides timestamp giving date, hour, minute at which the latest valid information on ChargingParkAvailability was provided.
if (bit 1 of selector is set)	
<IntUnLoMB>(minimalWaitingTime),	Minimal estimated time in minutes an end user will have to wait if there are no available charging places in this charging park.
if (bit 2 of selector is set)	
{	

<IntUnLoMB>(n),	
n *<ChargingStationAvailability>(chargingStationAvailability)	: For each ChargingStation of this ChargingPark, optionally specifies the available information. A charging park can contain multiple ChargingStations.
};	

A.2.2 ChargingStationAvailability

<ChargingStationAvailability>:=	
<IntUnLoMB>(stationID_Key),	: EMI internal identifier for a charging station. Within a charging park (identified by its parkID_Key), the stationID_Key uniquely identifies a charging station. The stationID_Key shall be used by an end-user device or application to combine charging station information, detailed charging station location and ChargingStationAvailability data and, optionally, the ReservationResponse.
<IntUnLoMB>(n),	
n *<FreePlacesForConnectorType>(freePlacesForConnectorType),	: Current information giving the number of free charging places for each ConnectorType at this charging station. This may be highly dynamic information.
BitArray(selector),	
if (bit 0 of selector is set)	
<IntUnTi>(minGuaranteedAmperage);	: The minimal amount of current in Ampere which is currently guaranteed by the charging park operator. This information can be used by an end user or end-user device to estimate charging time.

A.2.3 FreePlacesForConnectorType

<FreePlacesForConnectorType>:=	
<IntUnLoMB>(freePlaces),	: Current information giving the number of free charging places for this ConnectorType.
<emi002:ConnectorType>(connectorType),	: Connector type
BitArray(selector),	
if (bit 0 of selector is set)	

<emi003:Qualifier>(waitingTimeQualifier),	: For usage with waiting time only; qualifies the given waitingTime.
if (bit 1 of selector is set)	
<IntUnLoMB>(waitingTime);	: Estimated time in minutes an end user will have to wait if there are no available charging places at this ChargingStation. Can be quantified by means of the waitingTimeQuantifier.

A.2.4 ChargingParkSiteDescription

<ChargingParkSiteDescription>:=	
<ShortString>(parkName),	: Name of individual charging park.
<ShortString>(parkOperator),	: Name of operator of charging park.
BitArray(selector),	
if (bit 0 of selector is set)	
{	
<IntUnLoMB>(n),	
n *<OperatorContactInformation>(operatorContactInfo),	: Contact information for the operator of the charging park.
}	
if (bit 1 of selector is set)	
{	
<IntUnLoMB>(n),	
n *<LocalizedShortString>(parkAddress),	: Street address of the charging park in the relevant local language(s).
}	
if (bit 2 of selector is set)	
<Logo>(logo),	: Logo of the operator of the charging park.
if (bit 3 of selector is set)	
<ShortString>(providerExternalId),	: External identifier for the electromobility provider, if applicable (e.g. based on the EVSE ID).
if (bit 4 of selector is set)	
{	
<IntUnLoMB>(n),	

n *<ShortString>(roamingPartner),	: Information on electromobility providers which have roaming agreements with the operator of this charging park. Based on this information, an end user can verify if charging at this charging park is interesting from a financial point of view. In the case where the electromobility provider which the user has a contract with is not in the list of roamingPartner, use of this charging park might not be attractive.
}	
if (bit 5 of selector is set)	
<TimeToolkit>(openingHours),	: Opening hours of the charging park.
if (bit 6 of selector is set)	
{	
<IntUnLoMB>(n),	
n *<emi006:AssociatedServiceType>(associatedServices),	: Lists available services which the charging park offers in addition to charging facility (e.g. park-and-ride, access to public transportation, etc.).
}	
if (bit 7 of selector is set)	
{	
<IntUnLoMB>(n),	
n *<LocalizedShortString>(additionalDescription)	: Additional descriptive information on this charging park.
};	

A.2.5 Logo

<Logo>:=	
<ShortString>(mimeType),	: Mime type
BitArray(selector),	
if (bit 0 of selector is set)	
<ShortString>(src),	: Link (URL) to the logo
if (bit 1 of selector is set)	
<IntUnLoMB>(faviconLength),	: Length of favicon, max 1 024 bytes
if (bit 2 of selector is set)	
<LongString>(favicon);	: Contents of favicon, N bytes, max 1 024 bytes

A.2.6 OperatorContactInformation

<OperatorContactInformation>:=	
BitArray(selector),	
if (bit 0 of selector is set)	

<emi011:ContactType>(operatorContactType),	: Contact type
if (bit 1 of selector is set)	
<ShortString>(operatorContactText);	: Text containing phone number, etc.

A.2.7 SizeRestrictions

<SizeRestrictions>:=	
BitArray(selector),	
if (bit 0 of selector is set)	
<IntUnLoMB>(maxLength),	: Maximum length of the vehicle in centimetres.
if (bit 1 of selector is set)	
<IntUnLoMB>(maxWidth),	: Maximum width of the vehicle in centimetres.
if (bit 2 of selector is set)	
<IntUnLoMB>(maxHeight),	: Maximum height of the vehicle in centimetres.
if (bit 3 of selector is set)	
<Weight>(maxWeight),	: Maximum weight of the vehicle in kilogrammes.
if (bit 4 of selector is set)	
<IntUnLoMB>(minimalRequiredCableLength);	: Minimal required length of charging cable in centimetres.

A.2.8 PricingInformation

<PricingInformation>:=	
BitArray(selector),	
if (bit 0 of selector is set)	
<DateTime>(acquisitionTimeStamp),	: Time at which the last valid pricing information was published.
if (bit 1 of selector is set)	
<IntUnTi>(unitPriceResolution),	: Resolution of unit price (decimal points).
if (bit 2 of selector is set)	
<emi003:Qualifier>(priceQualifier),	: Qualification of the price information.
if (bit 3 of selector is set)	
<IntUnLoMB>(price),	: Price per delivery unit. Can be qualified by means of the qualifier.
if (bit 4 of selector is set)	
<emi001:BillingModel>(billingModel),	: The unit for which unit prices are given.
if (bit 5 of selector is set)	
<typ003:CurrencyType>(currencyType);	: Supported currency type(s) for payment at this charging park.

Annex B
(normative)

TPEG application, TPEG-ML representation

B.1 Message components

B.1.1 EMIMessage

```
<xs:element name=~"EMIMessage~" type=~"EMIMessage~"/>
<xs:complexType name=~"EMIMessage~">
 <xs:complexContent>
 <xs:extension base=~"tsf:ApplicationRootMessageML~">
 <xs:sequence>
 <xs:element name=~"mmt~" type=~"mmc:MessageManagementContainer~"/>
 <xs:element name=~"chargingParkAvailabilityVector~" type=~"ChargingParkAvailabilityVect
or~" minOccurs=~"0~" maxOccurs=~"unbounded~"/>
 <xs:element name=~"chargingParkInformation~" type=~"ChargingParkInformation~" minOccurs=~"0~"/>
 <xs:element name=~"reservationResponse~" type=~"ReservationResponse~" minOccurs=~"0~"/>
 <xs:element name=~"detailedChargingParkLocation~" type=~"DetailedChargingParkLocati
on~" minOccurs=~"0~"/>
 <xs:element name=~"chargingParkLocation~" type=~"lrc:LocationReferencingContainer~" minOccurs=~"0~"/>
 </xs:sequence>
 </xs:extension>
 </xs:complexContent>
</xs:complexType>
```

B.1.2 ChargingParkAvailabilityVector

```
<xs:complexType name=~"ChargingParkAvailabilityVector~">
 <xs:sequence>
 <xs:element name=~"timeStamp~" type=~"tdt:DateTime~"/>
 <xs:element name=~"chargingParkAvailability~" type=~"ChargingParkAvailabili
ty~" maxOccurs=~"unbounded~"/>
 <xs:element name=~"vectorLabel~" type=~"tdt:ShortString~" minOccurs=~"0~"/>
 </xs:sequence>
</xs:complexType>
```

B.1.3 ChargingParkInformation

```
<xs:complexType name=~”ChargingParkInformation~”>
 <xs:sequence>
 <xs:element name=~”parkID_Key~” type=~”tdt:IntUnLoMB~”/>
 <xs:element name=~”chargingParkSiteDescription~” type=~”ChargingParkSiteDescription~”/>
 <xs:element name=~”chargingParkCapacity~” type=~”tdt:IntUnLoMB~” minOccurs=~”0~”/>
 <xs:element name=~”electricityLabel~” type=~”tdt:ShortString~” minOccurs=~”0~” maxOccurs=~”unbou
nded~”/>
 <xs:element name=~”userType~” type=~”emi007_UserType~” minOccurs=~”0~” maxOccurs=~”unbounded~”/>
 <xs:element name=~”facilityType~” type=~”emi005_FacilityType~” minOccurs=~”0~”/>
 <xs:element name=~”reservability~” type=~”emi010_Reservability~” minOccurs=~”0~”/>
 <xs:element name=~”pricingInformation~” type=~”PricingInformation~” minOccurs=~”0~” maxOccurs=~”unb
ounded~”/>
 <xs:element name=~”paymentMethodType~” type=~”emi004_PaymentMethodType~” minOccurs=~”0~” maxOccurs=
~”unbounded~”/>
 <xs:element name=~”freeText~” type=~”tdt:LocalisedShortString~” minOccurs=~”0~” maxOccurs=~”unbou
nded~”/>
 <xs:element name=~”chargingStationInformation~” type=~”ChargingStationInformati
on~” minOccurs=~”0~” maxOccurs=~”unbounded~”/>
 </xs:sequence>
</xs:complexType>
```

B.1.4 ChargingStationInformation

```
<xs:complexType name=~”ChargingStationInformation~”>
 <xs:sequence>
 <xs:element name=~”stationID_Key~” type=~”tdt:IntUnLoMB~”/>
 <xs:element name=~”stationExternalId~” type=~”tdt:ShortString~” minOccurs=~”0~”/>
 <xs:element name=~”connectorType~” type=~”emi002_ConnectorType~” minOccurs=~”0~” maxOccurs=~”unbou
nded~”/>
 <xs:element name=~”stationType~” type=~”emi008_StationType~” minOccurs=~”0~”/>
 <xs:element name=~”vehicleType~” type=~”emi009_VehicleType~” minOccurs=~”0~” maxOccurs=~”unbounded~”/>
 <xs:element name=~”sizeRestrictions~” type=~”SizeRestrictions~” minOccurs=~”0~”/>
 </xs:sequence>
</xs:complexType>
```

B.1.5 DetailedChargingParkLocation

```
<xs:complexType name=~”DetailedChargingParkLocation~”>
 <xs:sequence>
 <xs:element name=~”parkID_Key~” type=~”tdt:IntUnLoMB~”/>
 <xs:element name=~”parkEntryLocation~” type=~”lrc:LocationReferencingContainer~” minOccurs=~”0~”/>
 <xs:element name=~”parkExitLocation~” type=~”lrc:LocationReferencingContainer~” minOccurs=~”0~”/>
 <xs:element name=~”detailedChargingStationLocation~” type=~”DetailedChargingStationLocati
on~” minOccurs=~”0~” maxOccurs=~”unbounded~”/>
 </xs:sequence>
</xs:complexType>
```

B.1.6 DetailedChargingStationLocation

```
<xs:complexType name=~"DetailedChargingStationLocation~">
 <xs:sequence>
 <xs:element name=~"stationID_Key~" type=~"tdt:IntUnLoMB~"/>
 <xs:element name=~"detailedStationLocation~" type=~"lrc:LocationReferencingContainer~"/>
 </xs:sequence>
</xs:complexType>
```

B.1.7 EMIMessageRequest

```
<xs:complexType name=~"EMIMessageRequest~">
 <xs:sequence>
 <xs:element name=~"mmt~" type=~"mmc:MessageManagementContainer~"/>
 <xs:element name=~"applicationInformation~" type=~"ReservationRequest~" minOccurs=~"0~"/>
 </xs:sequence>
</xs:complexType>
```

B.1.8 ReservationRequest

```
<xs:complexType name=~"ReservationRequest~">
 <xs:sequence>
 <xs:element name=~"authentificationId~" type=~"tdt:ShortString~"/>
 <xs:element name=~"paymentMethodType~" type=~"emi004_PaymentMethodType~"/>
 <xs:element name=~"connectorType~" type=~"emi002_ConnectorType~"/>
 <xs:element name=~"stationExternalId~" type=~"tdt:ShortString~" minOccurs=~"0~"/>
 <xs:element name=~"parkID_Key~" type=~"tdt:IntUnLoMB~" minOccurs=~"0~"/>
 <xs:element name=~"stationID_Key~" type=~"tdt:IntUnLoMB~" minOccurs=~"0~"/>
 <xs:element name=~"longitude~" type=~"tdt:IntSi24~" minOccurs=~"0~"/>
 <xs:element name=~"latitude~" type=~"tdt:IntSi24~" minOccurs=~"0~"/>
 <xs:element name=~"parkOperator~" type=~"tdt:ShortString~" minOccurs=~"0~"/>
 <xs:element name=~"providerExternalId~" type=~"tdt:ShortString~" minOccurs=~"0~"/>
 <xs:element name=~"vehicleId~" type=~"tdt:ShortString~" minOccurs=~"0~"/>
 <xs:element name=~"vehicleType~" type=~"emi009_VehicleType~" minOccurs=~"0~"/>
 <xs:element name=~"userType~" type=~"emi007_UserType~" minOccurs=~"0~"/>
 <xs:element name=~"estimatedArrivalTime~" type=~"tdt:DateTime~" minOccurs=~"0~"/>
 <xs:element name=~"estimatedPickupTime~" type=~"tdt:DateTime~" minOccurs=~"0~"/>
 <xs:element name=~"sizeRestrictions~" type=~"SizeRestrictions~" minOccurs=~"0~"/>
 </xs:sequence>
</xs:complexType>
```

B.1.9 ReservationResponse

```
<xs:complexType name=~”ReservationResponse~”>
 <xs:sequence>
 <xs:element name=~”reservationTimeStamp~” type=~”tdt:DateTime~”/>
 <xs:element name=~”reservationConfirmed~” type=~”tdt:Boolean~”/>
 <xs:element name=~”venueExternalId~” type=~”tdt:ShortString~” minOccurs=~”0~”/>
 <xs:element name=~”reservationId~” type=~”tdt:ShortString~” minOccurs=~”0~”/>
 <xs:element name=~”parkID_Key~” type=~”tdt:IntUnLoMB~” minOccurs=~”0~”/>
 <xs:element name=~”stationID_Key~” type=~”tdt:IntUnLoMB~” minOccurs=~”0~”/>
 <xs:element name=~”arrivalTime~” type=~”tdt:DateTime~” minOccurs=~”0~”/>
 <xs:element name=~”pickupTime~” type=~”tdt:DateTime~” minOccurs=~”0~”/>
 <xs:element name=~”reservationFreeText~” type=~”tdt:LocalisedShortString~” minOccurs=~”0~”/>
 </xs:sequence>
</xs:complexType>
```

B.2 Datatypes

B.2.1 ChargingParkAvailability

```
<xs:complexType name=~”ChargingParkAvailability~”>
 <xs:sequence>
 <xs:element name=~”parkID_Key~” type=~”tdt:IntUnLoMB~”/>
 <xs:element name=~”freePlacesForPark~” type=~”tdt:IntUnLoMB~”/>
 <xs:element name=~”timeStampForPark~” type=~”tdt:DateTime~” minOccurs=~”0~”/>
 <xs:element name=~”minimalWaitingTime~” type=~”tdt:IntUnLoMB~” minOccurs=~”0~”/>
 <xs:element name=~”chargingStationAvailability~” type=~”ChargingStationAvailabili
ty~” minOccurs=~”0~” maxOccurs=~”unbounded~”/>
 </xs:sequence>
</xs:complexType>
```

B.2.2 ChargingStationAvailability

```
<xs:complexType name=~”ChargingStationAvailability~”>
 <xs:sequence>
 <xs:element name=~”stationID_Key~” type=~”tdt:IntUnLoMB~”/>
 <xs:element name=~”freePlacesForConnectorType~” type=~”FreePlacesForConnectorTy
pe~” maxOccurs=~”unbounded~”/>
 <xs:element name=~”minGuaranteedAmperage~” type=~”tdt:IntUnTi~” minOccurs=~”0~”/>
 </xs:sequence>
</xs:complexType>
```

B.2.3 FreePlacesForConnectorType

```
<xs:complexType name=~”FreePlacesForConnectorType~”>
 <xs:sequence>
 <xs:element name=~”freePlaces~” type=~”tdt:IntUnLoMB~”/>
 <xs:element name=~”connectorType~” type=~”emi002_ConnectorType~”/>
 <xs:element name=~”waitingTimeQualifier~” type=~”emi003_Qualifier~” minOccurs=~”0~”/>
 <xs:element name=~”waitingTime~” type=~”tdt:IntUnLoMB~” minOccurs=~”0~”/>
 </xs:sequence>
</xs:complexType>
```

B.2.4 ChargingParkSiteDescription

```
<xs:complexType name=~”ChargingParkSiteDescription~”>
 <xs:sequence>
 <xs:element name=~”parkName~” type=~”tdt:ShortString~”/>
 <xs:element name=~”parkOperator~” type=~”tdt:ShortString~”/>
 <xs:element name=~”operatorContactInfo~” type=~”OperatorContactInformati
on~” minOccurs=~”0~” maxOccurs=~”unbounded~”/>
 <xs:element name=~”parkAddress~” type=~”tdt:LocalisedShortString~” minOccurs=~”0~” maxOccurs=~”unbo
unded~”/>
 <xs:element name=~”logo~” type=~”Logo~” minOccurs=~”0~”/>
 <xs:element name=~”providerExternalId~” type=~”tdt:ShortString~” minOccurs=~”0~”/>
 <xs:element name=~”roamingPartner~” type=~”tdt:ShortString~” minOccurs=~”0~” maxOccurs=~”unbounded~”/>
 <xs:element name=~”openingHours~” type=~”tdt:TimeToolkit~” minOccurs=~”0~”/>
 <xs:element name=~”associatedServices~” type=~”emi006_AssociatedServiceType~” minOccurs=~”0~” maxOc
curs=~”unbounded~”/>
 <xs:element name=~”additionalDescription~” type=~”tdt:LocalisedShortString~” minOccurs=~”0~” maxOcc
urs=~”unbounded~”/>
 </xs:sequence>
</xs:complexType>
```

B.2.5 Logo

```
<xs:complexType name=~”Logo~”>
 <xs:sequence>
 <xs:element name=~”mimeType~” type=~”tdt:ShortString~”/>
 <xs:element name=~”src~” type=~”tdt:ShortString~” minOccurs=~”0~”/>
 <xs:element name=~”faviconLength~” type=~”tdt:IntUnLoMB~” minOccurs=~”0~”/>
 <xs:element name=~”favicon~” type=~”tdt:LongString~” minOccurs=~”0~”/>
 </xs:sequence>
</xs:complexType>
```

B.2.6 OperatorContactInformation

```
<xs:complexType name=~”OperatorContactInformation~”>
 <xs:sequence>
 <xs:element name=~”operatorContactType~” type=~”emi011_ContactType~” minOccurs=~”0~”/>
 <xs:element name=~”operatorContactText~” type=~”tdt:ShortString~” minOccurs=~”0~”/>
 </xs:sequence>
</xs:complexType>
```

B.2.7 SizeRestrictions

```
<xs:complexType name=~"SizeRestrictions~">
 <xs:sequence>
 <xs:element name=~"maxLength~" type=~"tdt:IntUnLoMB~" minOccurs=~"0~"/>
 <xs:element name=~"maxWidth~" type=~"tdt:IntUnLoMB~" minOccurs=~"0~"/>
 <xs:element name=~"maxHeight~" type=~"tdt:IntUnLoMB~" minOccurs=~"0~"/>
 <xs:element name=~"maxWeight~" type=~"tdt:Weight~" minOccurs=~"0~"/>
 <xs:element name=~"minimalRequiredCableLength~" type=~"tdt:IntUnLoMB~" minOccurs=~"0~"/>
 </xs:sequence>
</xs:complexType>
```

B.2.8 PricingInformation

```
<xs:complexType name=~"PricingInformation~">
 <xs:sequence>
 <xs:element name=~"acquisitionTimeStamp~" type=~"tdt:DateTime~" minOccurs=~"0~"/>
 <xs:element name=~"unitPriceResolution~" type=~"tdt:IntUnTi~" minOccurs=~"0~"/>
 <xs:element name=~"priceQualifier~" type=~"emi003_Qualifier~" minOccurs=~"0~"/>
 <xs:element name=~"price~" type=~"tdt:IntUnLoMB~" minOccurs=~"0~"/>
 <xs:element name=~"billingModel~" type=~"emi001_BillingModel~" minOccurs=~"0~"/>
 <xs:element name=~"currencyType~" type=~"tdt:typ003_CurrencyType~" minOccurs=~"0~"/>
 </xs:sequence>
</xs:complexType>
```

B.3 Full EMI schema definition

```
<?xml version=~"1.0~" encoding=~"UTF-8~"?>
<!-- This XML schema is generated with tpegUMLconverter V2.3 -->
<xs:schema xmlns=~"http://www.tisa.org/TPEG/EMI_1_0~"
   targetNamespace=~"http://www.tisa.org/TPEG/EMI_1_0~"
   xmlns:xs=~"http://www.w3.org/2001/XMLSchema~"
   xmlns:tsf=~"http://www.tisa.org/TPEG/SFW_1_1~"
   xmlns:tdt=~"http://www.tisa.org/TPEG/TPEGDataTypes_2_0~"
   xmlns:mmc=~"http://www.tisa.org/TPEG/MMC_1_1~"
   xmlns:lrc=~"http://www.tisa.org/TPEG/LRC_2_1~"
   elementFormDefault=~"qualified~"
   attributeFormDefault=~"qualified~">
   <xs:import namespace=~"http://www.tisa.org/TPEG/SFW_1_1~" schemaLocation=~"SFW_1_1.xsd~"/>
   <xs:import namespace=~"http://www.tisa.org/TPEG/TPEGDataTypes_2_0~" schemaLocation=~"
TPEGDataTypes _2_0.xsd~"/>
```

```
   <xs:import namespace=~"http://www.tisa.org/TPEG/MMC_1_1~" schemaLocation=~"MMC_1_1.xsd~"/>
   <xs:import namespace=~"http://www.tisa.org/TPEG/LRC_2_1~" schemaLocation=~"LRC_2_1.xsd~"/>
   <xs:element name=~"EMIMessage~" type=~"EMIMessage~"/>
   <xs:complexType name=~"EMIMessage~">
      <xs:complexContent>
         <xs:extension base=~"tsf:ApplicationRootMessageML~">
            <xs:sequence>
               <xs:element name=~"mmt~" type=~"mmc:MessageManagementContainer~"/>
               <xs:element name=~"chargingParkAvailabilityVector~" type=~"ChargingParkAvailabilityVe
ctor~" minOccurs=~"0~" maxOccurs=~"unbounded~"/>
               <xs:element name=~"chargingParkInformation~" type=~"ChargingParkInformati
on~" minOccurs=~"0~"/>
              <xs:element name=~"reservationResponse~" type=~"ReservationResponse~" minOccurs=~"0~"/>
               <xs:element name=~"detailedChargingParkLocation~" type=~"DetailedChargingParkLocati
on~" minOccurs=~"0~"/>
               <xs:element name=~"chargingParkLocation~" type=~"lrc:LocationReferencingContainer~" m
inOccurs=~"0~"/>
            </xs:sequence>
         </xs:extension>
      </xs:complexContent>
   </xs:complexType>
   <xs:complexType name=~"ChargingParkAvailabilityVector~">
      <xs:sequence>
         <xs:element name=~"timeStamp~" type=~"tdt:DateTime~"/>
         <xs:element name=~"chargingParkAvailability~" type=~"ChargingParkAvailabili
ty~" maxOccurs=~"unbounded~"/>
         <xs:element name=~"vectorLabel~" type=~"tdt:ShortString~" minOccurs=~"0~"/>
      </xs:sequence>
   </xs:complexType>
   <xs:complexType name=~"ChargingParkInformation~">
      <xs:sequence>
         <xs:element name=~"parkID_Key~" type=~"tdt:IntUnLoMB~"/>
         <xs:element name=~"chargingParkSiteDescription~" type=~"ChargingParkSiteDescription~"/>
         <xs:element name=~"chargingParkCapacity~" type=~"tdt:IntUnLoMB~" minOccurs=~"0~"/>
         <xs:element name=~"electricityLabel~" type=~"tdt:ShortString~" minOccurs=~"0~" maxOccurs=~"
unbounded~"/>
          <xs:element name=~"userType~" type=~"emi007_UserType~" minOccurs=~"0~" maxOccurs=~"unbou
nded~"/>
         <xs:element name=~"facilityType~" type=~"emi005_FacilityType~" minOccurs=~"0~"/>
         <xs:element name=~"reservability~" type=~"emi010_Reservability~" minOccurs=~"0~"/>
         <xs:element name=~"pricingInformation~" type=~"PricingInformation~" minOccurs=~"0~" maxOccu
rs=~"unbounded~"/>
         <xs:element name=~"paymentMethodType~" type=~"emi004_PaymentMethodType~" minOccurs=~"0~" ma
xOccurs=~"unbounded~"/>
         <xs:element name=~"freeText~" type=~"tdt:LocalisedShortString~" minOccurs=~"0~" maxOccurs=~
"unbounded~"/>
         <xs:element name=~"chargingStationInformation~" type=~"ChargingStationInformati
on~" minOccurs=~"0~" maxOccurs=~"unbounded~"/>
      </xs:sequence>
   </xs:complexType>
```

```
   <xs:complexType name=~"ChargingStationInformation~">
      <xs:sequence>
         <xs:element name=~"stationID_Key~" type=~"tdt:IntUnLoMB~"/>
         <xs:element name=~"stationExternalId~" type=~"tdt:ShortString~" minOccurs=~"0~"/>
         <xs:element name=~"connectorType~" type=~"emi002_ConnectorType~" minOccurs=~"0~" maxOccurs=
~"unbounded~"/>
         <xs:element name=~"stationType~" type=~"emi008_StationType~" minOccurs=~"0~"/>
         <xs:element name=~"vehicleType~" type=~"emi009_VehicleType~" minOccurs=~"0~" maxOccurs=~"un
bounded~"/>
         <xs:element name=~"sizeRestrictions~" type=~"SizeRestrictions~" minOccurs=~"0~"/>
      </xs:sequence>
   </xs:complexType>
   <xs:complexType name=~"DetailedChargingParkLocation~">
      <xs:sequence>
         <xs:element name=~"parkID_Key~" type=~"tdt:IntUnLoMB~"/>
         <xs:element name=~"parkEntryLocation~" type=~"lrc:LocationReferencingContainer~" minOccurs
=~"0~"/>
         <xs:element name=~"parkExitLocation~" type=~"lrc:LocationReferencingContainer~" minOccurs
=~"0~"/>
         <xs:element name=~"detailedChargingStationLocation~" type=~"DetailedChargingStationLocati
on~" minOccurs=~"0~" maxOccurs=~"unbounded~"/>
      </xs:sequence>
   </xs:complexType>
   <xs:complexType name=~"DetailedChargingStationLocation~">
      <xs:sequence>
         <xs:element name=~"stationID_Key~" type=~"tdt:IntUnLoMB~"/>
         <xs:element name=~"detailedStationLocation~" type=~"lrc:LocationReferencingContainer~"/>
      </xs:sequence>
   </xs:complexType>
   <xs:complexType name=~"EMIMessageRequest~">
      <xs:sequence>
         <xs:element name=~"mmt~" type=~"mmc:MessageManagementContainer~"/>
         <xs:element name=~"applicationInformation~" type=~"ReservationRequest~" minOccurs=~"0~"/>
      </xs:sequence>
   </xs:complexType>
```

```
   <xs:complexType name=~”ReservationRequest~”>
      <xs:sequence>
         <xs:element name=~”authentificationId~” type=~”tdt:ShortString~”/>
         <xs:element name=~”paymentMethodType~” type=~”emi004_PaymentMethodType~”/>
         <xs:element name=~”connectorType~” type=~”emi002_ConnectorType~”/>
         <xs:element name=~”stationExternalId~” type=~”tdt:ShortString~” minOccurs=~”0~”/>
         <xs:element name=~”parkID_Key~” type=~”tdt:IntUnLoMB~” minOccurs=~”0~”/>
         <xs:element name=~”stationID_Key~” type=~”tdt:IntUnLoMB~” minOccurs=~”0~”/>
         <xs:element name=~”longitude~” type=~”tdt:IntSi24~” minOccurs=~”0~”/>
         <xs:element name=~”latitude~” type=~”tdt:IntSi24~” minOccurs=~”0~”/>
         <xs:element name=~”parkOperator~” type=~”tdt:ShortString~” minOccurs=~”0~”/>
         <xs:element name=~”providerExternalId~” type=~”tdt:ShortString~” minOccurs=~”0~”/>
         <xs:element name=~”vehicleId~” type=~”tdt:ShortString~” minOccurs=~”0~”/>
         <xs:element name=~”vehicleType~” type=~”emi009_VehicleType~” minOccurs=~”0~”/>
         <xs:element name=~”userType~” type=~”emi007_UserType~” minOccurs=~”0~”/>
         <xs:element name=~”estimatedArrivalTime~” type=~”tdt:DateTime~” minOccurs=~”0~”/>
         <xs:element name=~”estimatedPickupTime~” type=~”tdt:DateTime~” minOccurs=~”0~”/>
         <xs:element name=~”sizeRestrictions~” type=~”SizeRestrictions~” minOccurs=~”0~”/>
      </xs:sequence>
   </xs:complexType>
   <xs:complexType name=~”ReservationResponse~”>
      <xs:sequence>
         <xs:element name=~”reservationTimeStamp~” type=~”tdt:DateTime~”/>
         <xs:element name=~”reservationConfirmed~” type=~”tdt:Boolean~”/>
         <xs:element name=~”venueExternalId~” type=~”tdt:ShortString~” minOccurs=~”0~”/>
         <xs:element name=~”reservationId~” type=~”tdt:ShortString~” minOccurs=~”0~”/>
         <xs:element name=~”parkID_Key~” type=~”tdt:IntUnLoMB~” minOccurs=~”0~”/>
         <xs:element name=~”stationID_Key~” type=~”tdt:IntUnLoMB~” minOccurs=~”0~”/>
         <xs:element name=~”arrivalTime~” type=~”tdt:DateTime~” minOccurs=~”0~”/>
         <xs:element name=~”pickupTime~” type=~”tdt:DateTime~” minOccurs=~”0~”/>
        <xs:element name=~”reservationFreeText~” type=~”tdt:LocalisedShortString~” minOccurs=~”0~”/>
      </xs:sequence>
   </xs:complexType>
   <xs:complexType name=~”ChargingParkAvailability~”>
      <xs:sequence>
         <xs:element name=~”parkID_Key~” type=~”tdt:IntUnLoMB~”/>
         <xs:element name=~”freePlacesForPark~” type=~”tdt:IntUnLoMB~”/>
         <xs:element name=~”timeStampForPark~” type=~”tdt:DateTime~” minOccurs=~”0~”/>
         <xs:element name=~”minimalWaitingTime~” type=~”tdt:IntUnLoMB~” minOccurs=~”0~”/>
         <xs:element name=~”chargingStationAvailability~” type=~”ChargingStationAvailabili
ty~” minOccurs=~”0~” maxOccurs=~”unbounded~”/>
      </xs:sequence>
   </xs:complexType>
```

```
   <xs:complexType name=~"ChargingStationAvailability~">
      <xs:sequence>
         <xs:element name=~"stationID_Key~" type=~"tdt:IntUnLoMB~"/>
         <xs:element name=~"freePlacesForConnectorType~" type=~"FreePlacesForConnectorTy
pe~" maxOccurs=~"unbounded~"/>
         <xs:element name=~"minGuaranteedAmperage~" type=~"tdt:IntUnTi~" minOccurs=~"0~"/>
      </xs:sequence>
   </xs:complexType>
   <xs:complexType name=~"FreePlacesForConnectorType~">
      <xs:sequence>
         <xs:element name=~"freePlaces~" type=~"tdt:IntUnLoMB~"/>
         <xs:element name=~"connectorType~" type=~"emi002_ConnectorType~"/>
         <xs:element name=~"waitingTimeQualifier~" type=~"emi003_Qualifier~" minOccurs=~"0~"/>
         <xs:element name=~"waitingTime~" type=~"tdt:IntUnLoMB~" minOccurs=~"0~"/>
      </xs:sequence>
   </xs:complexType>
   <xs:complexType name=~"ChargingParkSiteDescription~">
      <xs:sequence>
         <xs:element name=~"parkName~" type=~"tdt:ShortString~"/>
         <xs:element name=~"parkOperator~" type=~"tdt:ShortString~"/>
         <xs:element name=~"operatorContactInfo~" type=~"OperatorContactInformati
on~" minOccurs=~"0~" maxOccurs=~"unbounded~"/>
         <xs:element name=~"parkAddress~" type=~"tdt:LocalisedShortString~" minOccurs=~"0~" maxOccur
s=~"unbounded~"/>
         <xs:element name=~"logo~" type=~"Logo~" minOccurs=~"0~"/>
         <xs:element name=~"providerExternalId~" type=~"tdt:ShortString~" minOccurs=~"0~"/>
         <xs:element name=~"roamingPartner~" type=~"tdt:ShortString~" minOccurs=~"0~" maxOccurs=~"un
bounded~"/>
         <xs:element name=~"openingHours~" type=~"tdt:TimeToolkit~" minOccurs=~"0~"/>
         <xs:element name=~"associatedServices~" type=~"emi006_AssociatedServiceType~" minOccurs=~"0
~" maxOccurs=~"unbounded~"/>
         <xs:element name=~"additionalDescription~" type=~"tdt:LocalisedShortString~" minOccurs=~"0~
" maxOccurs=~"unbounded~"/>
      </xs:sequence>
   </xs:complexType>
   <xs:complexType name=~"Logo~">
      <xs:sequence>
         <xs:element name=~"mimeType~" type=~"tdt:ShortString~"/>
         <xs:element name=~"src~" type=~"tdt:ShortString~" minOccurs=~"0~"/>
         <xs:element name=~"faviconLength~" type=~"tdt:IntUnLoMB~" minOccurs=~"0~"/>
         <xs:element name=~"favicon~" type=~"tdt:LongString~" minOccurs=~"0~"/>
      </xs:sequence>
   </xs:complexType>
```

```
<xs:complexType name=~"OperatorContactInformation~">
   <xs:sequence>
      <xs:element name=~"operatorContactType~" type=~"emi011_ContactType~" minOccurs=~"0~"/>
      <xs:element name=~"operatorContactText~" type=~"tdt:ShortString~" minOccurs=~"0~"/>
   </xs:sequence>
</xs:complexType>
<xs:complexType name=~"SizeRestrictions~">
   <xs:sequence>
      <xs:element name=~"maxLength~" type=~"tdt:IntUnLoMB~" minOccurs=~"0~"/>
      <xs:element name=~"maxWidth~" type=~"tdt:IntUnLoMB~" minOccurs=~"0~"/>
      <xs:element name=~"maxHeight~" type=~"tdt:IntUnLoMB~" minOccurs=~"0~"/>
      <xs:element name=~"maxWeight~" type=~"tdt:Weight~" minOccurs=~"0~"/>
      <xs:element name=~"minimalRequiredCableLength~" type=~"tdt:IntUnLoMB~" minOccurs=~"0~"/>
   </xs:sequence>
</xs:complexType>
<xs:complexType name=~"PricingInformation~">
   <xs:sequence>
      <xs:element name=~"acquisitionTimeStamp~" type=~"tdt:DateTime~" minOccurs=~"0~"/>
      <xs:element name=~"unitPriceResolution~" type=~"tdt:IntUnTi~" minOccurs=~"0~"/>
      <xs:element name=~"priceQualifier~" type=~"emi003_Qualifier~" minOccurs=~"0~"/>
      <xs:element name=~"price~" type=~"tdt:IntUnLoMB~" minOccurs=~"0~"/>
      <xs:element name=~"billingModel~" type=~"emi001_BillingModel~" minOccurs=~"0~"/>
      <xs:element name=~"currencyType~" type=~"tdt:typ003_CurrencyType~" minOccurs=~"0~"/>
   </xs:sequence>
</xs:complexType>
<xs:complexType name=~"emi001_BillingModel~">
   <xs:attribute name=~"table~" type=~"xs:string~" fixed=~"emi001_BillingModel~" use=~"required~"/>
   <xs:attribute name=~"code~" use=~"required~">
      <xs:simpleType>
         <xs:restriction base=~"xs:unsignedByte~">
            <xs:minInclusive value=~"0~"/>
            <xs:maxInclusive value=~"255~"/>
         </xs:restriction>
      </xs:simpleType>
   </xs:attribute>
</xs:complexType>
<xs:complexType name=~"emi002_ConnectorType~">
   <xs:attribute name=~"table~" type=~"xs:string~" fixed=~"emi002_ConnectorType~" use=~"required~"/>
   <xs:attribute name=~"code~" use=~"required~">
      <xs:simpleType>
         <xs:restriction base=~"xs:unsignedByte~">
            <xs:minInclusive value=~"0~"/>
            <xs:maxInclusive value=~"255~"/>
         </xs:restriction>
      </xs:simpleType>
   </xs:attribute>
</xs:complexType>
```

```
   <xs:complexType name=~"emi003_Qualifier~">
      <xs:attribute name=~"table~" type=~"xs:string~" fixed=~"emi003_Qualifier~" use=~"required~"/>
      <xs:attribute name=~"code~" use=~"required~">
         <xs:simpleType>
            <xs:restriction base=~"xs:unsignedByte~">
               <xs:minInclusive value=~"0~"/>
               <xs:maxInclusive value=~"255~"/>
            </xs:restriction>
         </xs:simpleType>
      </xs:attribute>
   </xs:complexType>
   <xs:complexType name=~"emi004_PaymentMethodType~">
      <xs:attribute name=~"table~" type=~"xs:string~" fixed=~"emi004_PaymentMethodType~" use=~"requ
ired~"/>
      <xs:attribute name=~"code~" use=~"required~">
         <xs:simpleType>
            <xs:restriction base=~"xs:unsignedByte~">
               <xs:minInclusive value=~"0~"/>
               <xs:maxInclusive value=~"255~"/>
            </xs:restriction>
         </xs:simpleType>
      </xs:attribute>
   </xs:complexType>
   <xs:complexType name=~"emi005_FacilityType~">
      <xs:attribute name=~"table~" type=~"xs:string~" fixed=~"emi005_FacilityType~" use=~"required~"/>
      <xs:attribute name=~"code~" use=~"required~">
         <xs:simpleType>
            <xs:restriction base=~"xs:unsignedByte~">
               <xs:minInclusive value=~"0~"/>
               <xs:maxInclusive value=~"255~"/>
            </xs:restriction>
         </xs:simpleType>
      </xs:attribute>
   </xs:complexType>
   <xs:complexType name=~"emi006_AssociatedServiceType~">
      <xs:attribute name=~"table~" type=~"xs:string~" fixed=~"emi006_AssociatedServiceType~" use=~"re
quired~"/>
      <xs:attribute name=~"code~" use=~"required~">
         <xs:simpleType>
            <xs:restriction base=~"xs:unsignedByte~">
               <xs:minInclusive value=~"0~"/>
               <xs:maxInclusive value=~"255~"/>
            </xs:restriction>
         </xs:simpleType>
      </xs:attribute>
   </xs:complexType>
```

```
<xs:complexType name=~"emi007_UserType~">
   <xs:attribute name=~"table~" type=~"xs:string~" fixed=~"emi007_UserType~" use=~"required~"/>
   <xs:attribute name=~"code~" use=~"required~">
      <xs:simpleType>
         <xs:restriction base=~"xs:unsignedByte~">
            <xs:minInclusive value=~"0~"/>
            <xs:maxInclusive value=~"255~"/>
         </xs:restriction>
      </xs:simpleType>
   </xs:attribute>
</xs:complexType>
<xs:complexType name=~"emi008_StationType~">
   <xs:attribute name=~"table~" type=~"xs:string~" fixed=~"emi008_StationType~" use=~"required~"/>
   <xs:attribute name=~"code~" use=~"required~">
      <xs:simpleType>
         <xs:restriction base=~"xs:unsignedByte~">
            <xs:minInclusive value=~"0~"/>
            <xs:maxInclusive value=~"255~"/>
         </xs:restriction>
      </xs:simpleType>
   </xs:attribute>
</xs:complexType>
<xs:complexType name=~"emi009_VehicleType~">
   <xs:attribute name=~"table~" type=~"xs:string~" fixed=~"emi009_VehicleType~" use=~"required~"/>
   <xs:attribute name=~"code~" use=~"required~">
      <xs:simpleType>
         <xs:restriction base=~"xs:unsignedByte~">
            <xs:minInclusive value=~"0~"/>
            <xs:maxInclusive value=~"255~"/>
         </xs:restriction>
      </xs:simpleType>
   </xs:attribute>
</xs:complexType>
<xs:complexType name=~"emi010_Reservability~">
   <xs:attribute name=~"table~" type=~"xs:string~" fixed=~"emi010_Reservability~" use=~"required~"/>
   <xs:attribute name=~"code~" use=~"required~">
      <xs:simpleType>
         <xs:restriction base=~"xs:unsignedByte~">
            <xs:minInclusive value=~"0~"/>
            <xs:maxInclusive value=~"255~"/>
         </xs:restriction>
      </xs:simpleType>
   </xs:attribute>
</xs:complexType>
```

```
   <xs:complexType name=~"emi011_ContactType~">
      <xs:attribute name=~"table~" type=~"xs:string~" fixed=~"emi011_ContactType~" use=~"required~"/>
      <xs:attribute name=~"code~" use=~"required~">
         <xs:simpleType>
            <xs:restriction base=~"xs:unsignedByte~">
               <xs:minInclusive value=~"0~"/>
               <xs:maxInclusive value=~"255~"/>
            </xs:restriction>
         </xs:simpleType>
      </xs:attribute>
   </xs:complexType>
</xs:schema>
```

Bibliography

[1] ISO/TS 21219-2, *Intelligent transport systems — Traffic and travel information (TTI) via transport protocol experts group, generation 2 (TPEG2) — Part 2: UML modelling rules*

[2] ISO/TS 21219-3, *Intelligent transport systems — Traffic and travel information (TTI) via transport protocol experts group, generation 2 (TPEG2) — Part 3: UML to binary conversion rules*

[3] ISO/TS 21219-4, *Intelligent transport systems — Traffic and travel information (TTI) via transport protocol experts group, generation 2 (TPEG2) — Part 4: UML to XML conversion rules*

[4] DIN SPEC 91286:2011, *Electric mobility — Schemes of identifiers for E-Roaming — Contract ID and Electric Vehicle Supply Equipment ID*

This page deliberately left blank

This page deliberately left blank

British Standards Institution (BSI)

BSI is the national body responsible for preparing British Standards and other standards-related publications, information and services.

BSI is incorporated by Royal Charter. British Standards and other standardization products are published by BSI Standards Limited.

About us

We bring together business, industry, government, consumers, innovators and others to shape their combined experience and expertise into standards -based solutions.

The knowledge embodied in our standards has been carefully assembled in a dependable format and refined through our open consultation process. Organizations of all sizes and across all sectors choose standards to help them achieve their goals.

Information on standards

We can provide you with the knowledge that your organization needs to succeed. Find out more about British Standards by visiting our website at bsigroup.com/standards or contacting our Customer Services team or Knowledge Centre.

Buying standards

You can buy and download PDF versions of BSI publications, including British and adopted European and international standards, through our website at bsigroup.com/shop, where hard copies can also be purchased.

If you need international and foreign standards from other Standards Development Organizations, hard copies can be ordered from our Customer Services team.

Copyright in BSI publications

All the content in BSI publications, including British Standards, is the property of and copyrighted by BSI or some person or entity that owns copyright in the information used (such as the international standardization bodies) and has formally licensed such information to BSI for commercial publication and use.

Save for the provisions below, you may not transfer, share or disseminate any portion of the standard to any other person. You may not adapt, distribute, commercially exploit, or publicly display the standard or any portion thereof in any manner whatsoever without BSI's prior written consent.

Storing and using standards

Standards purchased in soft copy format:

- A British Standard purchased in soft copy format is licensed to a sole named user for personal or internal company use only.
- The standard may be stored on more than 1 device provided that it is accessible by the sole named user only and that only 1 copy is accessed at any one time.
- A single paper copy may be printed for personal or internal company use only.

Standards purchased in hard copy format:

- A British Standard purchased in hard copy format is for personal or internal company use only.
- It may not be further reproduced – in any format – to create an additional copy. This includes scanning of the document.

If you need more than 1 copy of the document, or if you wish to share the document on an internal network, you can save money by choosing a subscription product (see 'Subscriptions').

Reproducing extracts

For permission to reproduce content from BSI publications contact the BSI Copyright & Licensing team.

Subscriptions

Our range of subscription services are designed to make using standards easier for you. For further information on our subscription products go to bsigroup.com/subscriptions.

With **British Standards Online (BSOL)** you'll have instant access to over 55,000 British and adopted European and international standards from your desktop. It's available 24/7 and is refreshed daily so you'll always be up to date.

You can keep in touch with standards developments and receive substantial discounts on the purchase price of standards, both in single copy and subscription format, by becoming a **BSI Subscribing Member**.

PLUS is an updating service exclusive to BSI Subscribing Members. You will automatically receive the latest hard copy of your standards when they're revised or replaced.

To find out more about becoming a BSI Subscribing Member and the benefits of membership, please visit bsigroup.com/shop.

With a **Multi-User Network Licence (MUNL)** you are able to host standards publications on your intranet. Licences can cover as few or as many users as you wish. With updates supplied as soon as they're available, you can be sure your documentation is current. For further information, email subscriptions@bsigroup.com.

Revisions

Our British Standards and other publications are updated by amendment or revision.

We continually improve the quality of our products and services to benefit your business. If you find an inaccuracy or ambiguity within a British Standard or other BSI publication please inform the Knowledge Centre.

Useful Contacts

Customer Services
Tel: +44 345 086 9001
Email (orders): orders@bsigroup.com
Email (enquiries): cservices@bsigroup.com

Subscriptions
Tel: +44 345 086 9001
Email: subscriptions@bsigroup.com

Knowledge Centre
Tel: +44 20 8996 7004
Email: knowledgecentre@bsigroup.com

Copyright & Licensing
Tel: +44 20 8996 7070
Email: copyright@bsigroup.com

BSI Group Headquarters

389 Chiswick High Road London W4 4AL UK

bsi.