TECHNICAL SPECIFICATION

ISO/TS 17575-1

First edition
2010-06-15

Electronic fee collection — Application interface definition for autonomous systems —

Part 1: **Charging**

Perception du télépéage — Définition de l'interface d'application pour les systèmes autonomes —

Partie 1: Imputation

Reference number
ISO/TS 17575-1:2010(E)

PDF disclaimer

This PDF file may contain embedded typefaces. In accordance with Adobe's licensing policy, this file may be printed or viewed but shall not be edited unless the typefaces which are embedded are licensed to and installed on the computer performing the editing. In downloading this file, parties accept therein the responsibility of not infringing Adobe's licensing policy. The ISO Central Secretariat accepts no liability in this area.

Adobe is a trademark of Adobe Systems Incorporated.

Details of the software products used to create this PDF file can be found in the General Info relative to the file; the PDF-creation parameters were optimized for printing. Every care has been taken to ensure that the file is suitable for use by ISO member bodies. In the unlikely event that a problem relating to it is found, please inform the Central Secretariat at the address given below.

COPYRIGHT PROTECTED DOCUMENT

ISO copyright office
Case postale 56 • CH-1211 Geneva 20
Tel. + 41 22 749 01 11
Fax + 41 22 749 09 47
E-mail copyright@iso.org
Web www.iso.org

Published in Switzerland

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.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 2.

The main task of technical committees is to prepare International Standards. Draft International Standards adopted by the technical committees are circulated to the member bodies for voting. Publication as an International Standard requires approval by at least 75 % of the member bodies casting a vote.

In other circumstances, particularly when there is an urgent market requirement for such documents, a technical committee may decide to publish other types of document:

— an ISO Publicly Available Specification (ISO/PAS) represents an agreement between technical experts in an ISO working group and is accepted for publication if it is approved by more than 50 % of the members of the parent committee casting a vote;

— an ISO Technical Specification (ISO/TS) represents an agreement between the members of a technical committee and is accepted for publication if it is approved by 2/3 of the members of the committee casting a vote.

An ISO/PAS or ISO/TS is reviewed after three years in order to decide whether it will be confirmed for a further three years, revised to become an International Standard, or withdrawn. If the ISO/PAS or ISO/TS is confirmed, it is reviewed again after a further three years, at which time it must either be transformed into an International Standard or be withdrawn.

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.

ISO/TS 17575-1 was prepared by the European Committee for Standardization (CEN) Technical Committee CEN/TC 278, *Road transport and traffic telematics,* in collaboration with Technical Committee ISO/TC 204, *Intelligent transport systems*, in accordance with the Agreement on technical cooperation between ISO and CEN (Vienna Agreement).

ISO/TS 17575 consists of the following parts, under the general title *Electronic fee collection — Application interface definition for autonomous systems*:

— *Part 1: Charging*

— *Part 2: Communication and connection to the lower layers*

— *Part 3: Context data*

— *Part 4: Roaming*

Introduction

Autonomous systems

This part of ISO/TS 17575 is part of a series of specifications defining the information exchange between the Front End and the Back End in Electronic Fee Collection (EFC) based on autonomous on-board equipment (OBE). EFC systems automatically collect charging data for the use of road infrastructure including motorway tolls, zone-based fees in urban areas, tolls for special infrastructure like bridges and tunnels, distance-based charging and parking fees.

Autonomous OBE operates without relying on dedicated road-side infrastructure by employing wide-area technologies such as Global Navigation Satellite Systems (GNSS) and Cellular Communications Networks (CN). These EFC systems are referred to by a variety of names. Besides the terms autonomous systems and GNSS/CN systems, also the terms GPS/GSM systems, and wide-area charging systems are in use.

Autonomous systems use satellite positioning, often combined with additional sensor technologies such as gyroscopes, odometers and accelerometers, to localize the vehicle and to find its position on a map containing the charged geographic objects, such as charged roads or charged areas. From the charged objects, the vehicle characteristics, the time of day and other data that are relevant for describing road use, the tariff and ultimately the road usage fee are determined.

Some of the strengths of the autonomous approach to electronic fee collection are its flexibility, allowing the implementation of almost all conceivable charging principles, and its independence from local infrastructure, thereby predisposing this technology towards interoperability across charging systems and countries. Interoperability can only be achieved with clearly defined interfaces, which is the aim and justification of ISO/TS 17575.

Business architecture

This part of ISO/TS 17575 complies with the business architecture defined in the draft of the future International Standard ISO 17573. According to this architecture, the Toll Charger is the provider of the road infrastructure and, hence, the recipient of the road usage charges. The Toll Charger is the actor associated with the Toll Charging role. See Figure 1.

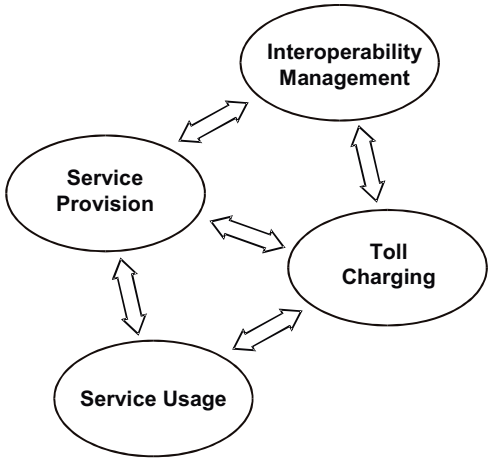

Figure 1 — The rolebased model underlying this Technical Specification

Service Providers issue OBE to the users of the road infrastructure. Service Providers are responsible for operating the OBE that will record the amount of road usage in all toll charging systems the vehicle passes through and for delivering the charging data to the individual Toll Chargers. In general, each Service Provider delivers charging data to several Toll Chargers, as well as each Toll Charger in general receives charging data from more than one Service Provider. Interoperability Management in Figure 1 comprises all specifications and activities that in common define and maintain a set of rules that govern the overall toll charging environment.

Technical architecture

The technical architecture of Figure 2 is independent of any particular practical realization. It reflects the fact that some processing functionalities can either be allocated to the OBE or to an associated off-board component (Proxy). An example of processing functionality that can be realized either on- or off-board is map-matching, where the vehicle locations in terms of measured coordinates from GNSS are associated to geographic objects on a map that either resides on- or off-board. Also tariffication can be done with OBE tariff tables and processing, or with an off-board component.

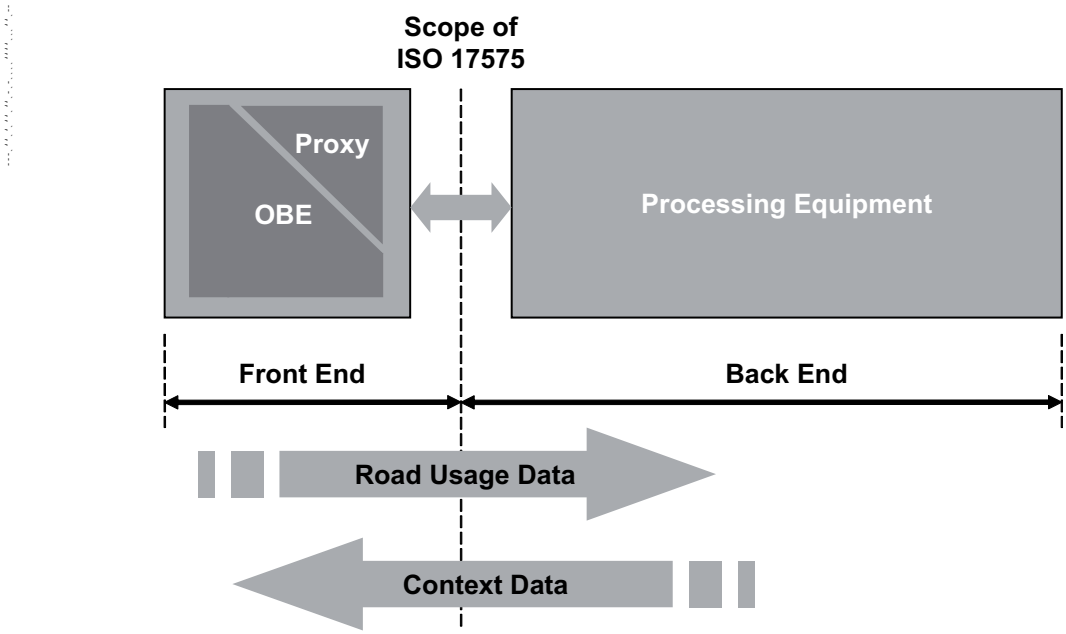

Figure 2 — Assumed technical architecture and interfaces

The combined functionality of OBE and Proxy is denoted as Front End. A Front End implementation where processing is predominately on OBE-side is known as a smart client (or intelligent client, fat client) or edge-heavy. A Front End where processing is mostly done off-board is denoted as thin-client or edge-light architecture. Many implementations between the “thin” and “thick” extremes are possible, as depicted by the gradual transition in the wedges in Figure 2. Both extremes of architectural choices have their merits and are one means where manufacturers compete with individual allocations of functionality between on-board and central resources.

Especially for thin client OBE, manufacturers might devise a wide variety of optimizations of the transfer of localization data between OBE and off-board components, where proprietary algorithms are used for data reduction and data compression. Standardization of this transfer is neither fully possible nor beneficial.

Location of the specification interface

In order to abstract from, and become independent of, these architectural implementation choices, the primary scope of ISO/TS 17575 is the data exchange between Front End and Back End (see the corresponding dotted line in Figure 2). For every toll regime, the Back End will send context data, i.e. a description of the toll regime in terms of charged objects, charging rules and, if required, the tariff scheme to the Front End, and will receive usage data from the Front End.

It has to be noted also that the distribution of tasks and responsibilities between Service Provider and Toll Charger will vary individually. Depending on the local legal situation, Toll Chargers will require “thinner” or “thicker” data, and might or might not leave certain data processing tasks to Service Providers. Hence, the data definitions in ISO/TS 17575 may be useful on several interfaces.

ISO/TS 17575 also provides for basic media-independent communication services that may be used for communication between Front End and Back End, which might be line-based or an air-link, and can also be used for the air-link between OBE and central communication server.

The parts of ISO/TS 17575

Part 1: Charging, defines the attributes for the transfer of usage data from the Front End to the Back End. The required attributes will differ from one Toll Charger to another, hence, attributes for all requirements are offered, ranging from attributes for raw localization data, for map-matched geographic objects and for completely priced toll transactions.

Part 2: Communication and connection to lower layers, defines basic communication services for data transfer over the OBE air-link or between Front End and Back End.

Part 3: Context Data, defines the data to be used for a description of individual charging systems in terms of charged geographical objects and charging and reporting rules. For every Toll Charger's system, attributes as defined in part 3 are used to transfer data to the Front End in order to instruct it which data to collect and report.

Part 4: Roaming, defines the functional details and data elements required to operate more than one EFC regime in parallel. The domains of these EFC regimes may or may not overlap. The charge rules of different overlapping EFC regimes can be linked, i.e. they may include rules that an area pricing scheme will not be charged if an overlapping toll road is used and already paid for.

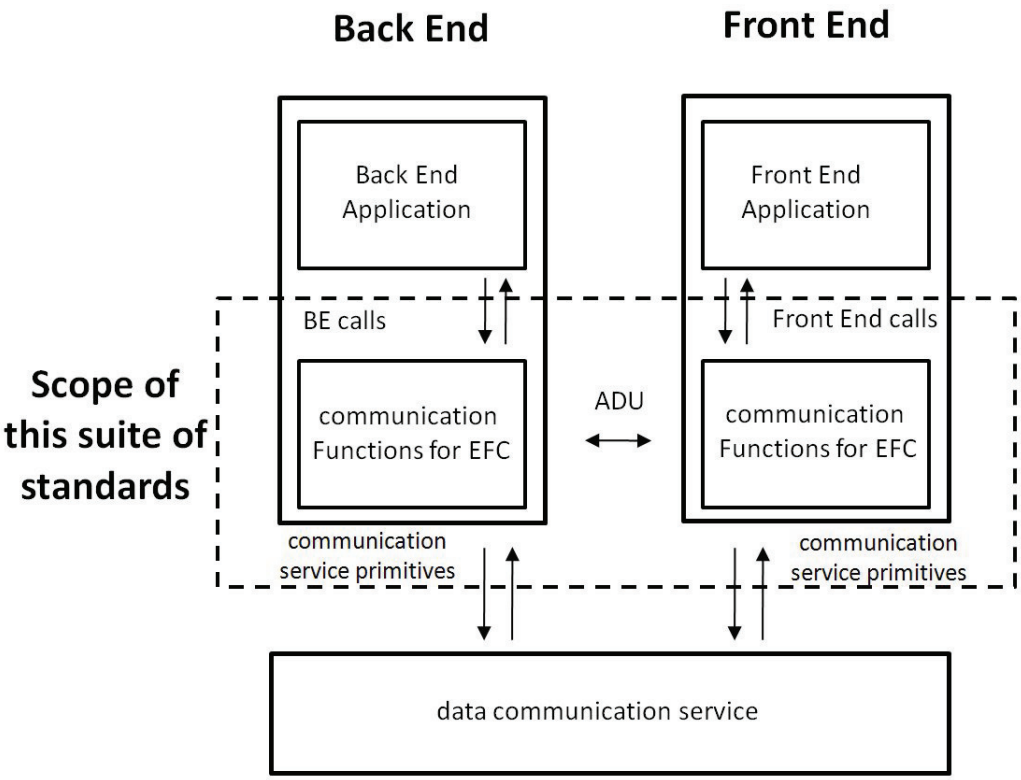

Figure 3 — Scope of ISO/TS 17575

Applicatory needs covered by ISO/TS 17575

- The parts of ISO/TS 17575 are compliant with the architecture defined in the future International Standard ISO 17573.
- The parts of ISO/TS 17575 support charges for use of road sections (including bridges, tunnels, passes, etc.), passage of cordons (entry/exit) and use of infrastructure within an area (distance, time).
- The parts of ISO/TS 17575 support fee collection based on units of distance or duration, and based on occurrence of events.
- The parts of ISO/TS 17575 support modulation of fees by vehicle category, road category, time of usage and contract type (e.g. exempt vehicles, special tariff vehicles, etc.).
- The parts of ISO/TS 17575 support limiting of fees by a defined maximum per period of usage.
- The parts of ISO/TS 17575 support fees with different legal status (e.g. public tax, private toll).
- The parts of ISO/TS 17575 support differing requirements of different Toll Chargers, especially in terms of
  - geographic domain and context descriptions,
  - contents and frequency of charge reports,
  - feedback to the driver (e.g. green or red light),
  - provision of additional detailed data on request, e.g. for settling of disputes.
- The parts of ISO/TS 17575 support overlapping geographic toll domains.
- The parts of ISO/TS 17575 support adaptations to changes in
  - tolled infrastructure,
  - tariffs, and
  - participating regimes.
- The parts of ISO/TS 17575 support the provision of trust guarantees by the Service Provider to the Toll Charger for the data originated from the Front End.

Electronic fee collection — Application interface definition for autonomous systems —

Part 1: Charging

1 Scope

This part of ISO/TS 17575 defines the format and semantic of the data exchange between a Front End (OBE plus optional proxy) and corresponding Back Ends in autonomous toll regimes. This part of ISO/TS 17575 deals with the definition of the data elements used to report charging details from the Front End to the Back End and to receive data which can be used to re-configure the ongoing process of gathering charge relevant information in the Front End.

The constitution of the charge report is dependent on configuration data that are assumed to be present in the Front End. The assembly of charge reports can be configured for each individual toll regime according to local needs. Charge reports generated in accordance with this part of ISO/TS 17575 are consistent with the requirements derived from the current architectural concept favoured in the relevant standardization bodies.

NOTE An EFC architecture standard is currently under development and is to be published in ISO 17573.

The data defined in this part of ISO/TS 17575 are used to generate charge reports that contain information about the road usage of a vehicle for certain time intervals. The contents of these charge reports might vary between toll regimes. A toll regime comprises a set of rules for charging, including the charged network, the charging principles, the liable vehicles and a definition of the required contents of the charge report.

The data defined in this part of ISO/TS 17575 are exchanged using an open definition of a communication stack as defined in ISO/TS 17575-2.

The definitions in this part of ISO/TS 17575 comprise:

— reporting data, i.e. data for transferring road usage data from Front End to Back End, including a response from the Back End towards the Front End;

— contract data, i.e. data for identifying contractually essential entities;

— road usage data, i.e. data for reporting the amount of road usage;

— account data for managing a payment account;

— versioning data;

— compliance checking data, i.e. data imported from ISO/TS 12813, which are required in Compliance Checking Communications.

2 Normative references

The following referenced documents are indispensable for the application 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 6709, *Standard representation of geographic point location by coordinates*

ISO/IEC 8824-1, *Information technology — Abstract Syntax Notation One (ASN.1): Specification of basic notation — Part 1*

ISO/IEC 8825-2, *Information technology — ASN.1 encoding rules: Specification of Packed Encoding Rules (PER) — Part 2*

ISO/TS 12813, *Electronic fee collection — Compliance check communication for autonomous systems*

ISO 14906, *Road transport and traffic telematics — Electronic fee collection — Application interface definition for dedicated short-range communication*

3 Terms and definitions

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

NOTE Some terms used in this document might also be defined in the future International Standard ISO 17573. The intention is to define them consistently. However, as ISO 17573 is still under development these definitions might be aligned in future.

3.1
area pricing
charging process based on road usage occurring within a given area

3.2
attribute
application information formed by one or by a sequence of data elements, used for implementation of a transaction

3.3
authenticator
data appended to, or a cryptographic transformation of, a data unit that allows a recipient of the data unit to prove the source and/or the integrity of the data unit and protect against forgery

[ISO 14906:2004, definition 3.4]

3.4
Back End
generic name for the computing and communication facilities of the Service Provider and/or the Toll Charger

3.5
charge report
data structure transmitted from the Front End to the Back End to report road usage data and supplementary related information

3.6
charge object
any object that is part of the toll context description that may be charged for its use under certain conditions

3.7
contract
agreement governing part of the collective behaviour of a set of objects

NOTE A contract specifies obligations, permissions and prohibitions for the objects involved.

3.8
cordon
border line of an area

3.9
cordon pricing
charging process based on registering passages of a cordon

3.10
data element
datum, which might itself consist of lower level data elements

3.11
data group
group of data elements selected by semantic relation

3.12
data integrity
property that data have not been altered or destroyed in an unauthorised manner

[ISO 14906:2004, definition 3.10]

3.13
Front End
part(s) of the toll system where road usage data for an individual road user are collected, processed and delivered to the Back End

NOTE The Front End comprises the on-board equipment and an optional proxy.

3.14
proxy
optional component of the Front End that communicates with on-board equipment and processes road usage data into a format compliant with this Technical Specification and delivers the data to the Back End

3.15
road
any stretch of land that can be navigated by a vehicle

3.16
road usage
travelling on a road with a vehicle

3.17
road usage data
data necessary to calculate the fees accumulated by a road user

3.18
road section tolling
processes for EFC based on charges for individual road sections

3.19
tarrification
calculation of the tariff

3.20
toll
charge, tax, fee or duty in connection with using a vehicle within a toll domain

NOTE The definition is the generalization of the classic definition of a toll as a charge, a tax, or a duty for permission to pass a barrier or to proceed along a road, over a bridge, etc. The definition above also includes fees regarded as an (administrative) obligation, e.g. a tax or a duty.

3.21
toll cluster
group of toll schemes operating under a common agreement providing interoperability for vehicles equipped with an appropriate OBE and being contracted under a service provider being part of the cluster

3.22
toll context
logical view of a toll scheme as defined by attributes and functions

3.23
toll context data
set of data necessary to define a toll context

3.24
toll domain
area or part of a road network where a toll regime is applied

3.25
toll regime
set of rules, including enforcement rules, governing the collection of toll in a toll

3.26
toll service
service enabling users having only one contract and one set of OBE to use a vehicle in one or more toll domains

3.27
toll system
overall view of a toll scheme or toll cluster

NOTE A component of a toll system can itself be a system, in which case it may be called a toll subsystem.

3.28
transaction
whole of the exchange of information between Front End and Back End necessary for the completion of a toll operation

3.29
transaction model
functional model describing the general structure of Electronic Payment Fee Collection transactions

[ISO 14906:2004, definition 3.20]

4 Abbreviations

For the purposes of this document, the following abbreviations apply unless otherwise specified.

— **ADU** Application data unit

— **ASN.1** Abstract Syntax Notation One (See ISO/IEC 8824-1.)

— **CCC** Compliance Check Communication, as defined by ISO/TS 12813

— **CN** Cellular network

— **DSRC** Dedicated short range communication

— **EFC** Electronic Fee Collection as defined in ISO 14906; here used as an equivalent to the term toll

— **GNSS** Global Navigation Satellite Systems

— **GPS** Global positioning system

— **GSM** Global System for Mobile Communications

— **HMI** Human-machine interface

— **OBE** On-board equipment

— **PICS** Protocol Implementation Conformance Statements

— **RSE** Road side equipment

— **VAT** Value added tax

5 Procedural requirements

5.1 General

This part of ISO/TS 17575 is intended to be used in autonomous toll systems set up according to an overall architecture currently favoured in the relevant standardization bodies.

NOTE An EFC architecture standard is currently under development and will be published as ISO 17573.

It defines the format and semantics of charge reports and charge report responses, which are part of the end-to-end information flow.

On-board equipment collects data on the road usage of an individual vehicle. These data are aggregated and processed regarding their relevance for charging either in the on-board equipment or in a proxy. The combination of on-board equipment and proxy is referred to as a Front End.

This part of ISO/TS 17575 defines the data required for communicating charge-relevant road usage data for an individual vehicle from the Front End to the Back End. The Front End shall accumulate road usage data into charge reports and send the charge reports to the Back End. The Back End shall confirm reception of a charge report (**`ChargeReport`**) with a charge report response (**`ChargeReportResponse`**).

5.2 Charge report configuration

All data elements comprising the attribute charge report are coded as optional (except for the **`usageStatementList`**, which ultimately also contains only optional elements).

For every toll regime, the Back End sends context data to the Front End. Context data is a description in terms of charge objects, charging rules and, if required, the tariff scheme.

Toll context data defines which data elements shall be present and which shall not. The Back End shall communicate the toll context data defining the requested charge report contents to the Front End before the on-board equipment is expected to collect road usage data. Upon reception of toll context data the Front End shall start to collect, process and accumulate road usage data into charge reports as requested. Toll context data also define upon which events charge reports shall be communicated.

NOTE 1 The charge report content requirements defined by the toll context data allow setting the report contents as required by the properties of the toll regime. These properties include the basic toll system types like:

— road section tolling (the charge relevant parameter is the sum of the road section lengths or tariff used by the vehicle);

— area pricing (the charge relevant parameter is either the distance driven inside the area or the time stayed inside the area);

— cordon pricing (the charge relevant parameter is the event of crossing the cordon around an area).

NOTE 2 Depending on local needs, Toll Chargers may require more or less processed data to varying levels of detail. Privacy considerations, enforcement approach and legal nature of the charge will also influence the choices agreed between toll charger and toll service provider regarding the requested contents of charge reports.

Charge reports support:

— reporting a list of charge objects that are declared as being used by the vehicle including associated tariff modifiers; this report may or may not include the calculated fee or tax;

— reports of road usage sessions within a single set of tariff modifiers; this report may or may not include the calculated fee or tax;

— report of contiguous sessions on a toll road or area where just the aggregated fee and the associated reference time is reported;

— reports where only the total fee within a predefined report period is forwarded (in this case it is anticipated that other means, outside the scope of this part of ISO/TS 17575, are used to allow a certain degree of validation of the charging process);

— any combination of the reports listed above.

5.3 Charge report response

The Back End shall respond to every received charge report with a charge report response. Which of the optional elements of the attribute **`ChargeReportResponse`** are present is not defined in this part of ISO/TS 17575.

NOTE The contents of the charge report response depend on the make and type of the Front End and on application software of the Front End and Back End as defined by the business requirements of the individual Service Provider. This part of ISO/TS 17575 only offers data elements for the response but does not impose restrictions upon the implementation and business choices by requiring mandatory content.

6 Data elements

6.1 Introduction

Data elements are grouped in logical groups for readability only.

The data group **Reporting** contains the main data elements of the charge report communication. These elements are the top level, overarching data structures containing all data elements described in this part of ISO/TS 17575.

The data group **General** contains data elements and types which are not explicitly part of other groups.

The data group **Contract** contains data elements and types related to road user contract information.

The data group **Usage** contains the information necessary to describe the usage of infrastructure causing eligibility for fees. These data are necessary for calculating the charges and for setting up correct bills and for settling disputes. The main data elements of this group present in the charge report and charge report response are respectively **`usageStatementList`** and **`dataReceived.`**

The data group **Account** contains the elements necessary to ensure that the correct account (and road user) is charged with the toll fees. The elements in the group Account are used for managing road user accounts in the Front End. These Front End accounts can contain the following types of data.

— Credit: the account holds a value corresponding to a monetary amount.

— Distance: the account holds a value representing a distance.

— Time: the account holds a value representing a point in time.

— Duration: the account holds a value representing a time duration.

— Event: the account holds a value representing a number of events.

The main data elements of this group present in the charge report and charge report response are respectively **`accountStatus`** and **`accountUpdate.`**

NOTE The kind of event counted in the respective option of the account data type is left to the implementation.

The data group **Versioning** contains data elements for version control of elements on the OBE.

The data group **Compliance Checking** provides information exchanged in compliance checking communication (CCC), as defined in ISO/TS 12813. Some of the data exchanged by CCC are already covered by other data elements, but for complete information about the content of CCC the data in this group are necessary.

6.2 Reporting

The two data types **`ChargeReport`** and **`ChargeReportResponse`** described in this subclause cover the complete charge report communication.

The data type **`ChargeReport`** comprises the following data elements:

— **`obeId,`**
— **`vehicleLPNr,`**
— **`paymentMeans,`**
— **`serviceProviderContract,`**
— **`tollCharger,`**
— **`timeOfReport,`**
— **`reportPeriod,`**
— **`versionInfo,`**
— **`usageStatementList,`**
— **`vatForThisSession,`**
— **`accountStatus,`**
— **`transactionCounter,`**
— **`mileage,`**
— **`cccAttributes,`**
— **`authenticator.`**

Those are the basic data elements for charge communication. A data element of the type **`ChargeReport`** is sent by the Front End whenever it is necessary to transmit charge data to the Back End.

The report contains the necessary data for identifying the (already registered) OBE and contract. It relays the information on usage of chargeable infrastructure by the vehicle and provides additional information for plausibility checks and accounting procedures (e.g. VAT calculations).

The data elements contained in this structure pertain to logical data groups and are detailed in the rest of Clause 6.

In response to a charge report, the Back End answers with a data element of the type **`ChargeReportResponse`**. This data type consists of the following components:

— **`reportRecipientId,`**
— **`dataReceived,`**
— **`versionsResponse,`**
— **`obeStatusForDriver,`**
— **`accountUpdate,`**
— **`responseAuthenticator.`**

These data provide a confirmation of the data reception at the application level. In addition feedback to the Front End (e.g. request for updates, change in OBE status) is provided.

The data types and elements contained in this structure and the ones constituting those pertain to logical data groups and are detailed below.

6.3 General

6.3.1 vehicleDescription

The data element **vehicleDescription** contains a list of characteristics (**vehicleLPNr**, **axles**, **class**, **dimensions**, **specificCharacteristics**, **ladenWeight**, **weightLimits**) of the vehicle. The data types of these elements are defined in and imported from ISO 14906.

6.3.2 timeOfReport

The data element **timeOfReport** gives the date and time when the charge report was compiled for transmission. The corresponding data type **DateAndTime** is defined in and imported from ISO 14906.

NOTE All data elements giving time information use the local time at the location of the vehicle.

6.3.3 reportPeriod

The data element **reportPeriod** gives the time period covered by the respective **ChargeReport**.

6.3.4 vatForThisSession

The data element **vatForThisSession** contains the aggregated VAT for the fees communicated in the respective charge report. Its associated data type is **PaymentFee,** which is defined in and imported from ISO 14906.

6.3.5 transactionCounter

The data element **transactionCounter** gives the number of the current charge report. This counter shall be incremented by the Front End after compilation of a charge report, facilitating distinction between charge reports. In the case of overflow the counter starts again at 0.

6.3.6 mileage

The data element **mileage** contains the reading of an internal mileage counter of type **Distance**. Counter reset and starting point shall be left to implementation; for one OBE and one contract the counter shall continuously count all vehicle mileage; counter shall start from zero in case of overflow (i.e. when reaching the maximum value).

6.3.7 Distance

The data type **Distance** contains distance values. The first element (**dist**) is an integer containing the distance value itself, the second (**disUnit**) is used for defining the unit of distance used. It can have the values **kilometres**, **miles** and **metres**.

6.3.8 Position

The data type **position** defines a geographical position, with the elements **longitude** and **latitude** as defined in ISO 6709.

6.3.9 Period

The data type **Period** defines a period of time, defined by the date and time of its beginning (**beginOfPeriod**) and of its end (**endOfPeriod**).

The data types of these elements are defined in and imported from ISO 14906.

6.3.10 Duration

The data type **Duration** defines a time span, defined by the actual value (**dur**) of the ASN.1 type **REAL**, and the respective unit (**durUnit**), which can have one of the values **seconds**, **minutes**, **hours**, **days** or **months**.

6.3.11 authenticator and responseAuthenticator

The data elements **authenticator** and **responseAuthenticator** are both of data type **MessageAuthenticator** and contain a security code intended for implementation of security measures. They can be used, for example, for ensuring integrity and authenticity of the respective data structure they are part of (through a message authentication code or signature on all the other data elements).

6.3.12 MessageAuthenticator

The data type **MessageAuthenticator** contains a security code of the ASN.1 type **OCTET STRING**, intended for implementation of security measures. It can be used, for example, for ensuring integrity and authenticity of the respective data structure of which they are part. The **MessageAuthenticator** always covers all data elements contained in the respective data structure.

NOTE The **UsageAuthenticator** in the data element **UsageStatement** is the only exception from the above rule. In this case only it can be used for authentication of data not contained in the respective **ChargeReport** and **ChargeReportResponse.**

The choice of specific algorithms for applying security measures is outside the scope of ISO/TS 17575.

6.4 Contract

6.4.1 obeId

The data element **obeId** is a unique identifier of the OBE. It contains two parts, one part (**manufacturerId**) a unique identification of the OBE manufacturer, and the other part (**EquipmentOBUId**) a manufacturer-specific identification of the individual OBE. The respective data types are defined in and imported from ISO 14906.

6.4.2 vehicleLPNr

The data element **vehicleLPNr** can be used either in the **chargeReport** data element or as part of the **vehicleDescription** in lower levels of the data structure. It shall only be present in the **chargeReport** if it is not used on the lower levels. The respective data type is defined in and imported from ISO 14906.

6.4.3 paymentMeans

The data element **paymentMeans** is a unique identification of an individual account, the respective data type is defined in and imported from ISO 14906.

6.4.4 serviceProviderContract

The data element **serviceProviderContract** identifies the service provider and the contract type to which the charge report data pertain. The respective data type is defined in and imported from ISO 14906.

NOTE 1 This is compatible with the concept favoured in the future architecture standard ISO 17573.

NOTE 2 It is the responsibility of the Service Provider to keep a consistent relationship between `obeId`, `paymentMeans` and `serviceProviderContract` in their database.

6.4.5 tollCharger

The data element **`tollCharger`** comprises two parts, identifying two of the sub roles of the toll charger:

— the toll operator of the respective toll scheme (data element **`efcOperator`**);

— the Transport Service Provider, recipient of fees collected in this scheme (data element **`recipient`**).

The data type **`Provider`** of both parts is imported from ISO 14906.

6.4.6 reportRecipientID

The data element **`reportRecipientID`** identifies the entity which received the respective usage statement.

6.4.7 obeStatusForDriver

The data element **`obeStatusForDriver`** contains information for controlling the HMI elements which communicate the status of the OBE and the contract to the driver. It is of the type **`ObeStatus`**.

6.4.8 ObeStatus

The data element **`ObeStatus`** contains information about the HMI elements that communicate the status of the OBE and the contract to the driver. The following values are foreseen:

— **`ok`,**
— **`nok`,**
— **`contactOperator`,**
— **`nokInLocalContext`,**
— **`noSignalling`.**

The value **`ok`** indicates that the OBE and contract are valid for toll collection in the current context; **`nok`** means that there is a problem and the road user has to undertake some action; **`contactOperator`** indicates that contacting the operator is expected from the road user; **`nokInLocalContext`** provides for the case that a local toll charger is not accepting the OBE or contract any longer, while it might still be valid in other contexts; **`noSignalling`** provides for the case that no indication whatsoever is given to the road user.

6.5 Usage

6.5.1 usageStatementList and UsageStatement

The data element **`usageStatementList`** contains a list of all the usage statements (data type **`UsageStatement`**) of the respective charge report.

The data type **`UsageStatement`** contains the information about actual road infrastructure use, which is required by the Back End to calculate the charges.

This data type consists of the following components:

— **`usageStatementID`**
— **`regimeID`**
— **`aggregatedFee`**
— **`aggregatedSingleTariffClassSession`**
— **`listOfChargeObjects`**
— **`listOfRawUsageData`**
— **`noUsage`**
— **`additionalUsageInformation`**
— **`usageAuthenticator`**

There are four options for describing road infrastructure use, which are represented by the following data elements:

- **aggregatedFee,**
- **aggregatedSingleTariffClassSession,**
- **listOfChargeObjects,**
- **listOfRawUsageData.**

Any combination of these options is possible. For special purposes (e.g. sign in) it is also possible to use empty usage statements. Those can be marked with the flag **noUsage.**

Security measures can be applied to the data type **UsageStatement** using the element **UsageAuthenticator**.

NOTE The data element **UsageAuthenticator** can also be used to implement the trusted recorder concept as described, for example, in the report of the European expert group 12. In this case the **UsageAuthenticator** does not contain a signature on the respective **UsageStatement**, but rather on data not contained in the respective **UsageStatement**.

6.5.2 usageStatementID

The data element **usageStatementID** is an identifier of the respective usage statement. The Front End shall assign this identifier and ensure its uniqueness within the charge report.

6.5.3 regimeID

The data element **regimeID** is an identifier of one of the regimes operated by a Toll Charger. This Toll Charger shall assign unique values within its realm.

6.5.4 aggregatedFee

The data element **aggregatedFee** contains the time period covered by the statement (**timePeriodCovered**) and the total amount of fee without VAT (**feeExclVat**) and the corresponding VAT **(vat)** aggregated within the time period given in **timePeriodCovered.**

6.5.5 aggregatedSingleTariffClassSession

The data element **aggregatedSingleTariffClassSession** contains the following data elements:

- **timePeriodCovered,**
- **vehicleDescription,**
- **tariffClass,**
- **totalDistanceCovered,**
- **numberOfDetectedEvents,**
- **ObeStatus,**
- **feeExclVat,**
- **vat**

It describes a single tariff class session, which is a part of a trip without any changes of charge-relevant parameters. Session changes may be due to changes in vehicle category, road category, time of day, etc. The information contained is: the time period covered by the statement (**timePeriodCovered**), the respective vehicle description (**vehicleDescription**) and tariff class (**tariffClass**), the total distance covered during the part of the trip (**totalDistanceCovered**), the number of events which occurred (**numberOfDetectedEvents**), the accumulated fee, excluding value added tax (**feeExclVat**) and the respective **vat**.

6.5.6 tariffClass

The data element **tariffClass** contains all information necessary for determining the tariff class in three data elements. The relevant parameters associated with those elements are the location class, the time class and

the user class. The location class (**locationClassId**), e.g. highway or city, is determined according to the local definition of the Toll Charger owning the respective toll scheme. The same is true for time class (**timeClassId**), e.g. day or weekend. The user class (**userClassId**) needed, for example, for identifying the driver as a handicapped person, is also a locally applicable definition.

6.5.7 listOfChargeObjects and DetectedChargeObject

The data element **listOfChargeObjects** contains a sequence of charge object descriptions (type **DetectedChargeObject**), each of which contains: the ID of the detected charge object (**chargeObjectId**).

The data element **DetectedChargeObject** also contains an additional number **subObjectNumber,** which identifies a sub-object within one and the same charge object. This is, for example, needed for cordons, where the cordon is the charge object, but there is a need for distinction between the various entry and exit points to this charge object. These cordon crossings are identified with sub-object numbers.

Furthermore, the data element **DetectedChargeObject** contains the time of passage of the charge object (**timeWhenUsed**), the reading of the internal virtual mileage counter at the time of use of the charge object (**mileageWhenUsed**), the respective vehicle description (**vehicleDescription**) and tariff class (**tariffClass**), the accumulated fee, excluding value added tax (**feeExclVat**) and the respective **vat**.

Additionally, charge objects can be marked as inferred or as detected with the support of a location augmentation beacon by setting the flag **chargeObjDetectionMode** accordingly.

An inferred charge object is one which was not detected by the evaluation of primary sensor data using regular rules defined for recognising charge objects, but inferred from the overall trip logic. In this case **chargeObjDetectionMode** is set to **inferred**.

For implementation reasons in special cases, the normal charge object detection technology could be supported (or even overruled) by localization augmentation beacons (LAC beacons), which communicate a location or even the passage of a given charge object directly, usually using short-range communication technology (e.g. DSRC). In this case **chargeObjDetectionMode** is set to **lac**.

6.5.8 ChargeObjectId

The data element **ChargeObjectId** identifies a charge object (e.g. road section, passage of cordon) according to the local definition of the Toll Charger owning the respective toll scheme. It contains the Toll Charger's identifier of the toll regime to which it belongs (**regimeId**), and a designation (**chargeObjectDesignation**).

NOTE The data element **regimeId** can be included either in a data element of type **UsageStatement** or in a data element of type **DetectedChargeObject**. The second option is only recommended in cases where more than one regime is covered in a single data element of type **UsageStatement**. Dealing with or avoiding contradictions resulting from use of this element on both levels simultaneously is left to the respective implementation.

6.5.9 ListOfRawUsageData

The data element **ListOfRawUsageData** contains three data elements: a list (**rawDataList**) containing a series of data elements (**measuredRawData**) of position (**measuredPosition**) and time (**timeWhenMeasured**), accompanied by the vehicle data (**vehicleDescription**) and tariff information (**tariffClass**) relevant for fee calculation.

NOTE ISO/TS 17575 does not define compression algorithms optimised for raw road usage data. Compression and data reduction may be used for data transfer, but are not supported at application level.

6.5.10 ListOfDSRCUsageData

The data element **listOfDSRCUsageData** itself is optional and contains a list of attributes (data type **AttributeList** which is defined in ISO 14906). The attribute list may be filled with EFC attributes that have been exchanged between OBE and RSE. The EFC attributes themselves are defined in ISO 14906.

NOTE The **listOfDSRCUsageData** element supports the use of autonomous Front Ends operated in DSRC based EFC domains. Usually the Toll Charger receives DSRC transactions from their RSE and transfers them at a certain point of time to the Service Provider for checking and justification. In addition to this, there might be a strong interest by the operator of the Back End to get a copy via the CN link of the data being sent to any RSE for performing consistency checks or update central accounts. In other scenarios there might be a requirement for autonomous Front Ends operating in DSRC based EFC domains independent of the RSE. They may generate a charge transaction in the same format as the DSRC transaction and forward it via the CN link to the Back End.

Due to the nature of the DSRC protocol (master-slave principle), it cannot be guaranteed that after a DSRC transaction has taken place both the Front End and the RSE hold the same application data. Therefore, it is assumed that for charging purposes only, data received and processed by the RSE are applicable and considered reliable.

6.5.11 additionalUsageInformation

The data element **additionalUsageInformation** can be used to transmit additional information needed by the Back End. This can, for example, be used to assign a usage statement to a certain cost centre for generating more detailed bills. The semantics and syntax of this data element are left to the respective implementation.

6.5.12 DataReceived

The data element **dataReceived** contains information about the data received in the charge report response. The respective data type **DataReceived** allows communicating the time of the report (through the data element **timeOfReport**), the corresponding value of the mileage counter (through the data element **mileage**), of the transaction counter (through the data element **transactionCounter**).

6.6 Account

6.6.1 accountStatus

With each charge report, the Front End has the option of communicating the status of the respective account. The respective data element **accountStatus** can have the following values:

- **ok**, i.e. contains a positive value above a defined threshold
- **low**, i.e. contains a positive value below a defined threshold
- **empty,** i.e. contains the value zero
- **negative**, i.e. contains a value below zero

NOTE The data element **accountStatus** is only relevant for implementations using on-board accounts.

6.6.2 accountUpdate

With each charge report response, the account in the Front End can be updated. The respective data element **accountUpdate** consist of three parts providing three options for updating the Front End account:

Either a predefined value is added to the current balance (through the data element **reloadAccount**), or the new balance is explicitly transmitted (through the data element **setAccount**). The third option adds a given quantity to the account (through the data element **addToAccount**). The update sets limits in credit, distance, time (defining a point of time until expiry), duration (defining duration until expiry) or a number of detected events.

6.6.3 ReloadAccount

The data element **reloadAccount** is of the type **ReloadAccount** which contains five Boolean data elements and the data element **reloadAuthenticator.** The Boolean values instruct the Front End application to either top up the respective type of account with a predefined value (**true**) or not (**false**). The predefined value shall be set in advance, either by the initial configuration process for the Front End or using an update mechanism and data elements.

Security measures can be applied to the data element **reloadAccount** using the element **reloadAuthenticator**.

6.6.4 NewAccountLimit

The data element **setAccount** is of type **NewAccountLimit,** which sets the account to a specific value. This data type contains five elements representing the new value for a specific type of account. The value is **newCreditLimit**, **newDistanceLimit**, **newTimeLimit**, **newDurationLimit**, or **newEventLimit**, depending on the type of the account.

Security measures can be applied to the data element **NewAccount** using the element **NewAuthenticator**.

6.6.5 AddToAccount

The data element **addToAccount** is of the type **AddToAccount,** which adds a specific value to the account. This data type contains five elements representing the value to be added to a specific type of account. This value is **addCredit**, **addDistance**, **addTime**, **addDuration**, or **addEvents**, depending on the type of the account.

Security measures can be applied to the data element **addToAccount** using the element **addAuthenticator**.

6.7 Versioning

6.7.1 versionInfo

The data element **versionInfo** indicates the current status of all relevant components of the Front End and shall be used to claim that all data, hardware and software versions in the charging process are up to date. It is of the type **VersionID,** which is equivalent to the type **OCTET STRING**. The coding, exact content and interpretation of this element is left to the respective implementation.

The content of this data element is defined by the actor responsible for updating the data for the Front End on application level. It holds the information for specifying the versions of all relevant hardware, software and data components relevant for updates. Therefore, it can be used for communicating the current status of versions in the Front End as well as for determining the need for updates to achieve a valid status for the Front End components.

6.7.2 versionResponse

The data element **versionResponse** indicates the due status of all relevant components of the Front End and shall be used to ensure that all data, hardware and software versions in the charging process are up to date. It either indicates that updates are necessary in the Front End for ensuring valid operation, or confirms the validity of versions of relevant hardware, software and data components.

It is of the type **VersionID,** which is equivalent to the type **octet string**. The coding, exact content and interpretation of this element is left to the respective implementation.

6.8 Compliance Checking — listOfCCCAttributes and CCCAttributes

The data element **listOfCCCAttributes** contains a sequence of data elements of the type **CCCAttributes**, distinguished by a time stamp.

The data in the structure **CCCAttributes** are defined in and imported from ISO/TS 12813. Elements of the type **CCCAttributes** contain information, which can be exchanged according to ISO/TS 12813, but is not covered by this part of ISO/TS 17575 elsewhere.

Annex A
(normative)

EFC data type specifications

```
ChargingModule {iso standard 17575 modules(0) efc(0) version(1)}

DEFINITIONS AUTOMATIC TAGS
::= BEGIN

IMPORTS

    EquipmentOBUId, PaymentMeans, EFC-ContextMark, Provider, DateAndTime,
    DriverCharacteristics, PaymentFee, VehicleAxles, VehicleClass, VehicleDimensions,
    VehicleLicencePlateNumber, VehicleSpecificCharacteristics, VehicleWeightLaden,
    VehicleWeightLimits,

FROM EfcModule {iso standard 14906 modules(0) efc(0) version(1)}

    AttributeList

FROM DSRCData {iso standard 14906 modules (0) dsrc (1) version (1)}


    VehicleAxlesHistory, CommunicationStatus, GnssStatus, DistanceRecordingStatus,
    ActiveContext, ObeHistory

FROM CccModule {iso standard 12813 modules(0) efc(0) version(1)};

-- NOTE: The following are the definitions of the EFC charge communication parameters

-------------------------------------------
--Level 1
-------------------------------------------

ChargeReport ::=      SEQUENCE {
    obeId                 ObeId OPTIONAL,
    vehicleLPNr           VehicleLicencePlateNumber OPTIONAL,
    paymentMeans          PaymentMeans OPTIONAL,        -- 14906
    serviceProviderContract   EFC-ContextMark,          -- 14906
    tollCharger           TollCharger OPTIONAL,
    timeOfReport          DateAndTime OPTIONAL,         -- 14906
    reportPeriod          Period,
    versionInfo           VersionID OPTIONAL,
    usageStatementList    SEQUENCE OF UsageStatement,
    vatForThisSession     PaymentFee OPTIONAL,          -- 14906
    accountStatus         AccountStatus OPTIONAL,
    transactionCounter    INTEGER (0..4294967295) OPTIONAL,
    mileage               Distance OPTIONAL,
    listOfCCCAttributes   SEQUENCE OF CCCAttributes OPTIONAL,
    authenticator         MessageAuthenticator OPTIONAL,
    ...
    }

ChargeReportResponse ::=  SEQUENCE {
    reportRecipientId     Provider OPTIONAL,            -- 14906
    dataReceived          DataReceived OPTIONAL,
    versionsResponse      VersionID OPTIONAL,
    obeStatusForDriver    OBEStatus OPTIONAL,           -- like 14906, new value (3)
    accountUpdate         AccountUpdate OPTIONAL,
    responseAuthenticator MessageAuthenticator OPTIONAL,
    ...
    }
```

```
-------------------------------------------
--Level 2
-------------------------------------------

ObeId ::=              SEQUENCE {
    manufacturerId         INTEGER(0..65535),     -- see ObeConfiguration in 14906
    equipmentOBUId         EquipmentOBUId         -- 14906
    }

TollCharger ::=        SEQUENCE {
    efcOperator            Provider,
    recipient              Provider
    }

UsageStatement ::=     SEQUENCE {
    usageStatementID       INTEGER (0..65535) OPTIONAL,
    regimeID               INTEGER (0..65535) OPTIONAL,
    aggregatedFee          AggregatedFee OPTIONAL,
    vat                    PaymentFee OPTIONAL,          -- 14906
    aggregatedSingleTariffClassSession   AggregatedSingleTariffClassSession OPTIONAL,
    listOfChargeObjects    SEQUENCE OF DetectedChargeObject OPTIONAL,
    listOfDSRCUsageData    ListOfDSRCUsageData OPTIONAL,
    listOfRawUsageData     ListOfRawUsageData OPTIONAL,
    noUsage                BOOLEAN OPTIONAL,
    additionalUsageInformation OCTET STRING OPTIONAL,
    usageAuthenticator     MessageAuthenticator OPTIONAL
    }

AccountStatus ::=      ENUMERATED {
    ok                     (0),
    low                    (1),
    empty                  (2),
    negative               (3),
    ...
    }

VersionID ::=          OCTET STRING

DataReceived::=        CHOICE {
    timeOfReport           DateAndTime,
    mileage                Distance,
    transactionCounter     INTEGER
    }

OBEStatus ::=          ENUMERATED {                  -- values like SetMMIRq in 14906,
    ok                     (0),                      -- value (3) added.
    nok                    (1),
    contactOperator        (2),
    nokInLocalContext      (3),
    noSignalling           (255)
    }

AccountUpdate ::=      CHOICE {
    reloadAccount          ReloadAccount,
    setAccount             NewAccountLimit,
    addToAccount           AddToAccount
    }

Distance ::=           SEQUENCE {
    dist                   INTEGER(0..16777215),
    disUnit                DisUnit DEFAULT kilometres
    }

MessageAuthenticator ::=  OCTET STRING

CCCAttributes ::=      SEQUENCE {
    timeOfCCCRecord        DateAndTime OPTIONAL,         -- 14906
```

```
    axlesHistory          VehicleAxlesHistory OPTIONAL,
    commStatus            CommunicationStatus OPTIONAL,
    gnssStatus            GnssStatus OPTIONAL,
    distRecStatus         DistanceRecordingStatus OPTIONAL,
    activeContext         ActiveContext OPTIONAL,
    obeHistory            ObeHistory OPTIONAL
    }

-------------------------------------------
--Level 3
-------------------------------------------

AggregatedFee ::=     SEQUENCE {
    timePeriodCovered     Period,
    feeExclVat            PaymentFee,
    vat                   PaymentFee OPTIONAL
    }

AggregatedSingleTariffClassSession ::=   SEQUENCE {
    timePeriodCovered     Period OPTIONAL,
    vehicleDescription    VehicleDescription OPTIONAL,
    tariffClass           TariffClass OPTIONAL,
    totalDistanceCovered  Distance OPTIONAL,
    numberOfDetectedEvents INTEGER OPTIONAL,
    obeStatus             OBEStatus OPTIONAL,       -- like 14906, new value (3),
    feeExclVat            PaymentFee OPTIONAL,      -- 14906
    vat                   PaymentFee OPTIONAL       -- 14906
    }

DetectedChargeObject ::=  SEQUENCE {
    chargeObjectId        ChargeObjectId,
    subObjectNumber       INTEGER (0..4294967295) OPTIONAL,
    timeWhenUsed          DateAndTime OPTIONAL,
    mileageWhenUsed       Distance OPTIONAL,
    vehicleDescription    VehicleDescription OPTIONAL,
    tariffClass           TariffClass OPTIONAL,
    obeStatus             OBEStatus OPTIONAL,       -- like 14906, new value (3),
    feeExclVat            PaymentFee OPTIONAL,      -- 14906
    vat                   PaymentFee OPTIONAL,      -- 14906
    chargeObjDetectionMode    DetectionMode OPTIONAL
    }

ListOfDSRCUsageData ::=   AttributeList      -- EN ISO 14906
                                             -- will contain EFC attributes as defined in
                                             -- 14906


ListOfRawUsageData ::=SEQUENCE {
    rawDataList           SEQUENCE OF MeasuredRawData,
    vehicleDescription    VehicleDescription OPTIONAL,
    tariffClass           TariffClass OPTIONAL
    }

ReloadAccount ::=     SEQUENCE {
    reloadOldCreditAmount BOOLEAN OPTIONAL,
    reloadOldDistanceLimit BOOLEAN OPTIONAL,
    reloadOldTimeLimit    BOOLEAN OPTIONAL,
    reloadOldDurationLimit BOOLEAN OPTIONAL,
    reloadOldEventLimit   BOOLEAN OPTIONAL,
    reloadAuthenticator   MessageAuthenticator OPTIONAL,
    ...
    }

NewAccountLimit ::=   SEQUENCE {
    newCreditLimit        PaymentFee OPTIONAL,      -- 14906
    newDistanceLimit      Distance OPTIONAL,        -- on toll roads or general
    newTimeLimit          DateAndTime OPTIONAL,
    newDurationLimit      Duration OPTIONAL,
```

```
    newEventLimit          INTEGER OPTIONAL,         -- number of events till next report
    newAuthenticator       MessageAuthenticator OPTIONAL,
    ...
    }

AddToAccount ::=       SEQUENCE {
    addCredit              PaymentFee OPTIONAL,      -- 14906
    addDistance            Distance OPTIONAL,        -- on toll roads or general
    addTime                DateAndTime OPTIONAL,
    addDuration            Duration OPTIONAL,
    addEvents              INTEGER OPTIONAL,         -- number of events till next report
    addAuthenticator       MessageAuthenticator OPTIONAL,
    ...
    }

Period ::=     SEQUENCE {
    beginOfPeriod      DateAndTime,
    endOfPeriod        DateAndTime
    }

DisUnit::=             ENUMERATED {
    kilometres             (0),
    miles                  (1),
    metres                 (2),
    ...
    }

-----------------------------------------
--Level 4
-----------------------------------------

VehicleDescription ::=SEQUENCE {
    vehicleLPNr        VehicleLicencePlateNumber OPTIONAL,
    axles              VehicleAxles OPTIONAL,
    class              VehicleClass OPTIONAL,
    dimensions         VehicleDimensions OPTIONAL,
    specificCharacteristics   VehicleSpecificCharacteristics OPTIONAL,
    ladenWeight        VehicleWeightLaden OPTIONAL,
    weightLimits       VehicleWeightLimits OPTIONAL,
    ...
    }

TariffClass ::=    SEQUENCE {
    locationClassId        INTEGER OPTIONAL,      -- local definition
    timeClassId            INTEGER OPTIONAL,      -- local definition
    userClassId            INTEGER  OPTIONAL   -- local definition
    }

ChargeObjectId ::=     SEQUENCE {
    regimeId                       INTEGER (0..65535) OPTIONAL,
    chargeObjectDesignation        INTEGER (0..4294967295)
    }

GetStampedTransactions ::=    SEQUENCE {
   getStampedRq                    GetStampedRq,
   getStampedRs                    GetStampedRs
   }

MeasuredRawData ::=    SEQUENCE {
    measuredPosition       Position,
    timeWhenMeasured       DateAndTime,
    ...
    }

Duration ::=           SEQUENCE {
    dur                    REAL,
    durUnit                DurUnit DEFAULT seconds
    }
```

```
-------------------------------------------
--Level 5
-------------------------------------------

Position ::=           SEQUENCE {
    longitude              INTEGER(-2147483648..2147483647), -- as defined in ISO 6709
                                                             -- in microdegrees, >0=east,
                                                             -- <0=west, absolute value <=180°
    latitude               INTEGER(-2147483648..2147483647) -- as defined in ISO 6709
                                                             -- in microdegrees, >0=north,
                                                             -- <0=south, absolute value <=90°
    }

DetectionMode ::=      ENUMERATED {
    measured               (0),
    inferred               (1),
    lac                    (2)
    }

DurUnit ::=            ENUMERATED {
    seconds                (0),
    minutes                (1),
    hours                  (2),
    days                   (3),
    months                 (4),
    ...
    }

END
```

Annex B
(normative)

PICS proforma

B.1 Introduction

This annex contains the Protocol Implementation Conformance Statements (PICS) proforma to be used for Front End implementation of the charge report protocol defined in Clause 6 and Annex A. The PICS proforma for the Back End are exactly the same, so for implementing both sides of the interface two sets of the tables below are necessary.

B.2 Mandatory data types

Table B.1 — Mandatory data types

Element	Status	Implemented
ChargeReport	mandatory	Yes/No
ChargeReportResponse	mandatory	Yes/No
UsageStatement	mandatory	Yes/No

B.3 Optional data elements

B.3.1 ChargeReport

Table B.2 — ChargeReport

Element	Status	Implemented
obeId	optional	Yes/No
vehicleLPNr	optional	Yes/No
paymentMeans	optional	Yes/No
serviceProviderContract	mandatory	Yes/No
tollCharger	optional	Yes/No
timeOfReport	optional	Yes/No
versionInfo	optional	Yes/No
usageStatementList	mandatory	Yes/No
vatForThisSession	optional	Yes/No
accountStatus	optional	Yes/No
transactionCounter	optional	Yes/No
mileage	optional	Yes/No
listOfCCCAttributes	optional	Yes/No
authenticator	optional	Yes/No

B.3.2 ChargeReportResponse

Table B.3 — ChargeReportResponse

Element	Status	Implemented
reportRecipientId	optional	Yes/No
dataReceived	optional	Yes/No
versionsResponse	optional	Yes/No
obeStatusForDriver	optional	Yes/No
accountUpdate	optional	Yes/No
responseAuthenticator	optional	Yes/No

B.3.3 UsageStatement

Table B.4 — UsageStatement

Element	Status	Implemented
usageStatementID	optional	Yes/No
regimeID	optional	Yes/No
aggregatedFee	optional	Yes/No
vat	optional	Yes/No
aggregatedSingleTariffClassSession	optional	Yes/No
listOfChargeObjects	optional	Yes/No
listOfDSRCUsageData	optional	Yes/No
listOfRawUsageData	optional	Yes/No
noUsage	optional	Yes/No
additionalUsageInformation	optional	Yes/No
usageAuthenticator	optional	Yes/No

B.3.4 Account types

Table B.5 — Account types

Account type	Status	Implemented
Credit	optional	Yes/No
Distance	optional	Yes/No
Time	optional	Yes/No
Duration	optional	Yes/No
Events	optional	Yes/No

Annex C
(informative)

Hierarchical data structure illustration

The data types and associated coding related to the data elements described in Clause 6 are defined using the Abstract Syntax Notation One (ASN.1) technique in accordance with ISO/IEC 8824-1. The packed encoding rules in accordance with ISO/IEC 8825-2 shall be applied. Data elements defined in lower levels are part of data elements in higher levels, e.g. the data elements of level 3 are part of the data elements in level 1 or 2. This hierarchy of data elements is shown in Figure C.1. Only the major elements are shown, which is also the reason for level 5 being empty in Figure C.1.

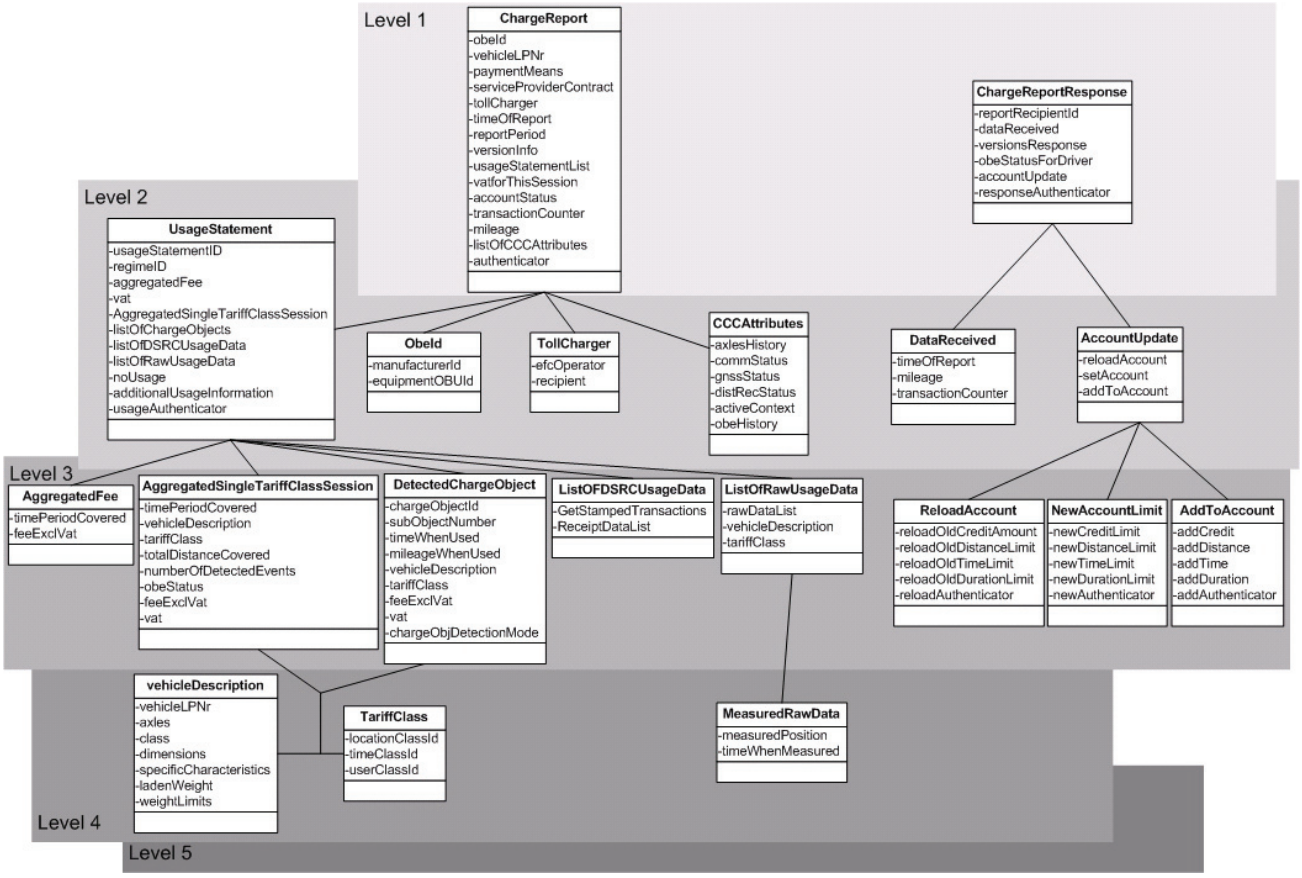

Figure C.1 — Hierarchy of data elements

Bibliography

[1] ISO/IEC 9646-7, *Information technology — Open Systems Interconnection — Conformance testing methodology and framework — Part 7: Implementation Conformance Statements*

[2] ISO 17573:—[1)], *Electronic fee collection — Systems architecture for vehicle related tolling*

[3] ISO/TS 17575-3:— [1)], *Electronic fee collection — Application interface definition for autonomous systems — Part 3: Context data*

[4] ISO/TS 17575-4:—[1)], *Electronic fee collection — Application interface definition for autonomous systems — Part 4: Roaming*

1) To be published.

ICS 03.220.20; 35.240.60

Price based on 23 pages