

BS EN ISO 17575-3:2016



BSI Standards Publication

Electronic fee collection — Application interface definition for autonomous systems

Part 3: Context data (ISO 17575-3:2016)

bsi.

...making excellence a habit.
AVA Documents™
Document Sharing Hub

National foreword

This British Standard is the UK implementation of EN ISO 17575-3:2016. It supersedes DD CEN ISO/TS 17575-3:2011 which is withdrawn.

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.

© The British Standards Institution 2016. Published by BSI Standards Limited 2016

ISBN 978 0 580 84630 4

ICS 03.220.20; 35.240.60

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

This British Standard was published under the authority of the Standards Policy and Strategy Committee on 29 February 2016.

Amendments issued since publication

Date	Text affected
------	---------------

English Version

Electronic fee collection - Application interface definition
for autonomous systems - Part 3: Context data (ISO 17575-
3:2016)

Perception du télépéage - Définition de l'interface
d'application pour les systèmes autonomes - Partie 3:
Données du contexte (ISO 17575-3:2016)

Elektronische Gebührenerhebung - Definition der
Anwendungsschnittstelle für autonome Systeme - Teil
3: Kontextdaten (ISO 17575-3:2016)

This European Standard was approved by CEN on 11 December 2015.

CEN members are bound to comply with the CEN/CENELEC Internal Regulations which stipulate the conditions for giving this European Standard the status of a national standard without any alteration. Up-to-date lists and bibliographical references concerning such national standards may be obtained on application to the CEN-CENELEC Management Centre or to any CEN member.

This European Standard exists in three official versions (English, French, German). A version in any other language made by translation under the responsibility of a CEN member into its own language and notified to the CEN-CENELEC Management Centre has the same status as the official versions.

CEN members are the national standards bodies of Austria, Belgium, Bulgaria, Croatia, Cyprus, Czech Republic, Denmark, Estonia, Finland, Former Yugoslav Republic of Macedonia, France, Germany, Greece, Hungary, Iceland, Ireland, Italy, Latvia, Lithuania, Luxembourg, Malta, Netherlands, Norway, Poland, Portugal, Romania, Slovakia, Slovenia, Spain, Sweden, Switzerland, Turkey and United Kingdom.



EUROPEAN COMMITTEE FOR STANDARDIZATION
COMITÉ EUROPÉEN DE NORMALISATION
EUROPÄISCHES KOMITEE FÜR NORMUNG

CEN-CENELEC Management Centre: Avenue Marnix 17, B-1000 Brussels

European foreword

This document (EN ISO 17575-3:2016) has been prepared by Technical Committee ISO/TC 204 “Intelligent transport systems” in collaboration with Technical Committee CEN/TC 278 “Intelligent transport systems” the secretariat of which is held by NEN.

This European Standard shall be given the status of a national standard, either by publication of an identical text or by endorsement, at the latest by August 2016, and conflicting national standards shall be withdrawn at the latest by August 2016.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. CEN [and/or CENELEC] shall not be held responsible for identifying any or all such patent rights.

This document supersedes CEN ISO/TS 17575-3:2011.

This document has been prepared under a mandate given to CEN by the European Commission and the European Free Trade Association.

According to the CEN-CENELEC Internal Regulations, the national standards organizations of the following countries are bound to implement this European Standard: Austria, Belgium, Bulgaria, Croatia, Cyprus, Czech Republic, Denmark, Estonia, Finland, Former Yugoslav Republic of Macedonia, France, Germany, Greece, Hungary, Iceland, Ireland, Italy, Latvia, Lithuania, Luxembourg, Malta, Netherlands, Norway, Poland, Portugal, Romania, Slovakia, Slovenia, Spain, Sweden, Switzerland, Turkey and the United Kingdom.

Endorsement notice

The text of ISO 17575-3:2016 has been approved by CEN as EN ISO 17575-3:2016 without any modification.

Contents

Page

Foreword	iv
Introduction	vi
1 Scope	1
2 Normative references	2
3 Terms and definitions	2
4 Abbreviated terms	4
5 General concept and overview	5
6 Procedural requirements and encoding rules	7
6.1 General	7
6.2 Communication services	7
6.3 Version and validity handling	7
6.3.1 Protocol versioning	7
6.3.2 Context data versioning	7
6.4 Encoding rules	8
6.5 Acknowledgement and behaviour on errors	8
7 Application data units	8
7.1 General	8
7.2 Message authentication (data type Iso17575-3-InformationContent)	9
7.3 Application data unit structure (data type Iso17575-3Adu)	9
7.4 Application data unit header (data type ISO 17575-3AduHeader)	10
7.5 Application data unit body (data type ISO 17575-3AduBody)	11
8 EFC Attributes	11
8.1 General	11
8.2 Rules with respect to support of context data	12
8.3 Attributes and data sets	12
8.4 EFC attributes authentication	12
8.5 EFC attributes data catalogue	13
8.5.1 General	13
8.5.2 Requirements with regards to context overview	14
8.5.3 Requirements with regards to tariff information	17
8.5.4 Requirements with regards to context layout	35
8.5.5 Requirements with regards to reporting rules	45
Annex A (normative) Data type specifications	59
Annex B (normative) Protocol implementation conformance statement (PICS) proforma	60
Annex C (informative) Hierarchical data structure illustration	98
Annex D (informative) How to use context data to define the properties of an EFC regime	103
Annex E (informative) Guidelines on the use of standardised digital maps in GDF format in the description of section based toll context layouts	108
Annex F (informative) Examples using EFC context data for scheme definitions	111
Annex G (informative) Use of this part of ISO 17575 for the EETS	116
Bibliography	118

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 WTO principles in the Technical Barriers to Trade (TBT) see the following URL: Foreword - Supplementary information

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

This edition of ISO 17575-3 cancels and replaces ISO/TS 17575-3:2011, which has been technically revised. The following changes have been made:

- conversion from a Technical Specification to an International Standard;
- amendments to reflect changes to the underlying base standards, especially ISO 14906;
- major changes regarding
 - integration of functionalities for the support of complex toll domains that consist of more than one partition from ISO/TS 17575-2:2010,
 - changes in the security scheme details,
 - introduction of protocol version identification,
 - harmonization of the identification of toll contexts amongst the parts of ISO 17575,
 - improvement of the possibility to use rounding rules,
 - enabling the use of a second alternative currency in tariffs,
 - adaptation of the charge reporting configuration to changes in ISO 17575-1:2016,
 - enabling the use of toll context partitions which may be present in one single toll context,
 - support of optional geographic data files (GDF) based description of toll liable networks (embracing such data definitions from ISO 12855:2012,
 - revised terms and definitions ([Clause 3](#)), and
 - editorial and formal corrections as well as changes to improve readability.

ISO 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*

In this edition of the ISO 17575-series the contents of ISO/TS 17575-4:2011 were incorporated into ISO 17575-3:2016. ISO/TS 17575-4:2011 will be withdrawn once ISO 17575-3 has been published.

Introduction

0.1 Autonomous systems

ISO 17575 is a series of standards 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 such as bridges and tunnels, distance-based charging, and parking fees.

Further introductory explanations of autonomous systems in EFC and, in particular, the considerations with respect to business and technical architecture that form the base for interfaces within such system and their interoperable specification are provided in ISO 17575-1:2016.

0.2 Location of the specification interface

In order to abstract from, and become independent of, these architectural implementation choices, the primary scope of ISO 17575 is the data exchange between Front End and Back End (see the corresponding vertical line in [Figure 1](#)). For every toll scheme, the Back End will send context data, i.e. a description of the toll scheme 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 17575 may be useful on several interfaces.

ISO 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.

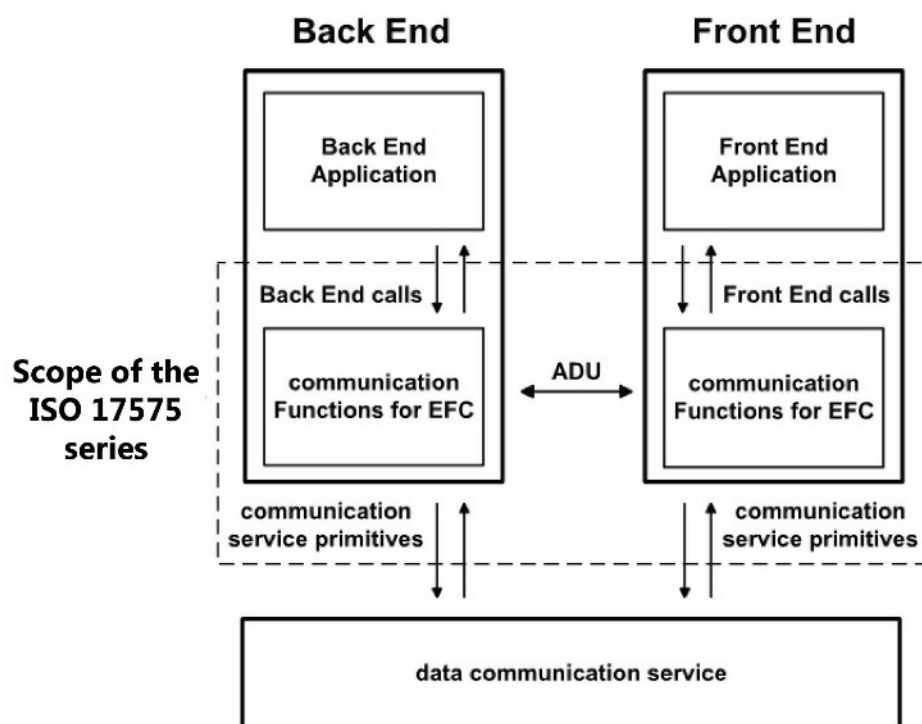


Figure 1 — Scope of ISO 17575

0.3 The parts of ISO 17575

Part 1: Charging, defines the attributes for the transfer of usage data from the Front End to the Back End. The contents of charge reports might vary between toll regimes, 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. 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.

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. The data defined in ISO 17575-1 and ISO 17575-3 can, but need not be, exchanged using the communication stack as defined in ISO 17575-2.

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 ISO 17575-3 are used to transfer data to the Front End in order to instruct it on which data to collect and report.

0.4 Application needs covered by ISO 17575

The ISO 17575-series of standards

- is compliant with the architecture defined in ISO 17573:2010,
- supports 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),
- supports fee collection based on units of distance or duration, and based on occurrence of events,
- supports modulation of fees by vehicle category, road category, time of usage and contract type (e.g. exempt vehicles, special tariff vehicles, etc.),
- supports limiting of fees by a defined maximum per period of usage,
- supports fees with different legal status (e.g. public tax, private toll),
- supports 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), and
 - provision of additional detailed data on request, e.g. for settling of disputes,
- supports overlapping geographic toll domains,
- supports adaptations to changes in
 - tolled infrastructure,
 - tariffs, and
 - participating toll schemes, and
- supports 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 3: Context data

1 Scope

This part of ISO 17575 defines the content, semantics and format of the data exchange between a Front End (OBE plus optional proxy) and the corresponding Back End in autonomous toll systems. It defines the data elements used to specify and describe the toll context details. Context data are transmitted from the Back End to the Front End to configure it for the charging processes of the associated toll context.

In ISO 17575, context data is the description of the properties of a single instance of an electronic fee collection (EFC) context. This single instance of an EFC context operates according to one of the basic tolling principles such as

- road section charging,
- area charging (according to travelled distance or duration of time), and
- cordon charging.

EFC context data comprise a set of rules for charging, including the description of the charged network, the charging principles, the liable vehicles and a definition of the required contents of the charge report. This set of rules is defined individually for each EFC context according to local needs.

The following data and associated procedures are defined in this part of ISO 17575:

- data providing toll context overview information;
- data providing tariff information (including definitions of required tariff determinants such as vehicle parameters, time classe, etc.);
- data providing context layout information;
- data providing reporting rules information.

This part of ISO 17575 also provides the required definitions and data specifications to be applied when one single toll context is split into more than one toll context partitions. This is applicable to cases where one EFC scheme and the rules applied cannot be described with a single set of context data.

[Annex A](#) provides the data type specification using ASN.1 notation.

The protocol implementation conformity statements (PICS) proforma are provided in [Annex B](#).

[Annex C](#) provides a graphical presentation of the structure of the toll context data.

[Annexes D, E and F](#) contain further information and descriptions, which may support the understanding and the implementation of the rules specified in this part of ISO 17575.

[Annex G](#) provides information how this part of ISO 17575 can be used in a European Electronic Toll Service (EETS) environment, with reference to EU Decision 2009/750.

2 Normative references

The following documents, in whole or in part, are normatively referenced in this document and are indispensable for its application. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO 612, *Road vehicles — Dimensions of motor vehicles and towed vehicles — Terms and definitions*

ISO 1176, *Road vehicles — Masses — Vocabulary and codes*

ISO 4217, *Codes for the representation of currencies and funds*

ISO/IEC 8824-1:2008, *Information technology — Abstract Syntax Notation One (ASN.1): Specification of basic notation — Part 1*

ISO/IEC 8825-2:2008, *Information technology — ASN.1 encoding rules: Specification of Packed Encoding Rules (PER) — Part 2*

ISO 12813:2015, *Electronic fee collection — Compliance check communication for autonomous systems*

ISO 14906:2011/Amd1:2015, *Electronic fee collection — Application interface definition for dedicated short-range communication*

ISO 17575-1:2016, *Electronic fee collection — Application interface definition for autonomous systems — Part 1: Charging*

EN 15509:2014, *Electronic fee collection — Interoperability application profile for DSRC*

NIMA TR8350.2, Third Edition — Amendment 1, January 2000, Department of Defense — *World Geodetic System 1984, Its Definition and Relationships With Local Geodetic Systems, issued by National Imagery and Mapping Agency (NIMA)*, US Department of Defense

3 Terms and definitions

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

- 3.1**
area charging
charging based on road usage within a given area
[SOURCE: ISO 17575-1:2016, 3.1]
- 3.2**
attribute
addressable package of data consisting of a single data element or structured sequences of data elements
[SOURCE: ISO 17575-1:2016, 3.2]
- 3.3**
authenticator
data, possibly encrypted, that is used for authentication
[SOURCE: EN 15509:2014, 3.3]
- 3.4**
Back End
part of a back office system interfacing to one or more *Front Ends* ([3.11](#))
[SOURCE: ISO 17575-1:2016, 3.4]

3.5

charge object

geographic or road related object for the use of which a charge is applied

[SOURCE: ISO 17575-1:2016, 3.5]

3.6

charge report

information containing road usage and related information originated at the *Front End* ([3.11](#))

[SOURCE: ISO 17575-1:2016, 3.6]

3.7

cordon

border line of an area

[SOURCE: ISO 17575-1:2016, 3.7]

3.8

cordon charging

charging for the crossing of a *cordon* ([3.7](#))

[SOURCE: ISO 17575-1:2016, 3.8]

3.9

data element

coded information, which might itself consist of lower level information structures

[SOURCE: ISO 17575-1:2016, 3.9]

3.10

data set

logical set of *data elements* ([3.9](#)) with a semantic relation

Note 1 to entry: Data set is used only for better understanding and is fully independent from implementation solutions.

3.11

Front End

part of a tolling system consisting of an *OBE* ([3.13](#)) and possibly a *proxy* ([3.14](#)) where road tolling information and usage data are collected and processed for delivery to the *Back End* ([3.4](#))

[SOURCE: ISO/TS 19299:2015, 3.17]

Note 1 to entry: The Front End comprises the *on-board equipment* ([3.13](#)) and an optional *proxy* ([3.14](#)).

3.12

layout

technical description of the location of tolled objects including their borders

3.13

on-board equipment

OBE

all required equipment on-board a vehicle for performing required EFC functions and communication services

3.14

proxy

optional part of a *Front End* ([3.11](#)) that communicates with external equipment and processes the data received into an agreed format to be delivered to the *Back End* ([3.4](#))

[SOURCE: ISO 17575-1:2016, 3.13]

3.15

road section charging

tolling principle where the fee is due if predefined sections of roads are used

[SOURCE: ISO 17575-1:2016, 3.14]

3.16

toll

charge, tax or duty levied in connection with using a vehicle in a *toll domain* (3.20)

[SOURCE: ISO/TS 19299:2015, 3.42, modified — “any” has been deleted from before “charge”.]

Note 1 to entry: 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 also includes fees regarded as an (administrative) obligation, e.g. a tax or a duty.

3.17

toll area

geographic area where a *toll* (3.16) is charged for road usage

3.18

toll context

logical view as defined by *attributes* (3.2) and functions of the basic elements of a toll scheme consisting of a single basic tolling principle, a spatial distribution of the *charge objects* (3.5) and a single behaviour of the related *Front End* (3.11)

[SOURCE: ISO 17575-1:2016, 3.17]

3.19

toll context data

information defined by the responsible toll charger as necessary to establish the *toll* (3.16) due for using a vehicle on a particular *toll context* (3.18) and to conclude the toll transaction

[SOURCE: ISO 12855:2015, 3.15]

3.20

toll domain

area or part of a road network where a certain *toll regime* (3.21) is applied

[SOURCE: ISO 17573:2010, 3.18, modified — “certain” has been added.]

3.21

toll regime

set of rules, including enforcement rules, governing the collection of *toll* (3.16) in a *toll domain* (3.20)

[SOURCE: ISO 17573:2010, 3.20]

3.22

toll scheme

organizational view of a *toll regime* (3.21), including the actors and their relationships

4 Abbreviated terms

For the purposes of this document, the following abbreviated terms apply.

ADU	Application data unit (ISO 14906)
ASN.1	Abstract Syntax Notation One (ISO/IEC 8824-1)
CCC	Compliance check communication (ISO 12813)

CN	Cellular network
DSRC	Dedicated short-range communication (ISO 14906)
DST	Daylight saving time
EFC	Electronic fee collection (ISO 14906)
GDF	Geographic Data Files (ISO 14825)
GNSS	Global Navigation Satellite Systems
HOT	High occupancy tolling
ID	Identifier
OBE	On-board equipment
PICS	Protocol implementation conformance statements
UTC	Coordinated Universal Time
VAT	Value added tax

5 General concept and overview

To enable a Front End to operate autonomously in a toll domain in the expected manner, a particular set of data elements containing application data has to be available to the Front End. These data elements shall contain a description of the rules that apply in a toll domain. This includes information regarding tariffs, vehicle classes, description of the charge objects, etc.

The data elements may be made available to the Front End using the communication services described in ISO 17575-2:2016.

For the purpose of data transfer an application data unit (ADU) is defined, which comprises a header (mainly containing identification and data management information) and a data body (containing the application data elements itself).

The ADU header allows for identification of the data originator and the data sender. Furthermore, it contains information about the toll context to which the application data belong. Finally, the ADU header carries a sequence number.

This part of ISO 17575 is based on the assumption that one toll scheme may consist of multiple parts. The data requirements provided in this part of ISO 17575 support this concept. In addition, Front Ends may be used in more than one toll scheme. In such cases, the Front End might have the capability to manage multiple sets of toll context data elements (one per toll scheme). See [Figure 2](#).

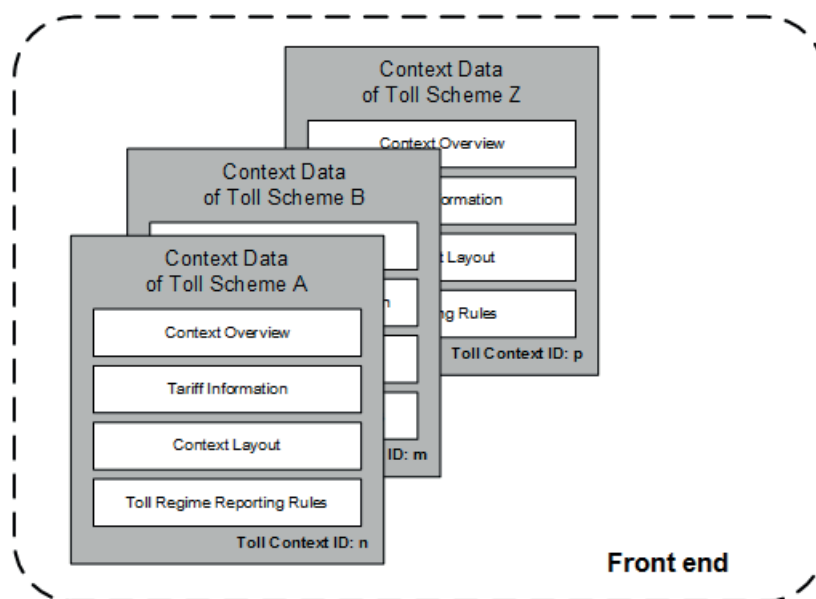


Figure 2 — Logical structure of toll context descriptions in a Front End

There may be a maximum number of toll contexts a Front End can manage. This number may depend on the memory size, the complexity of the toll context data and the envisaged use of the Front End. Front Ends may also be designed in a way to support the context description for one particular toll scheme only. Other Front End designs may support context descriptions for more than one toll scheme.

Context data are structured into logical data sets (see 8.3). Figure 3 gives an overview of these data sets and the type of information belonging to each data set.

Each data set comprises one or more EFC attributes. EFC attributes contain the application data. They are defined in Clause 8.

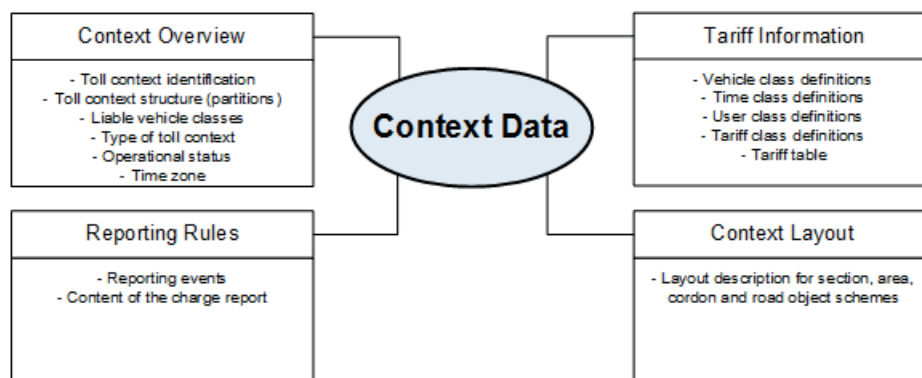


Figure 3 — Context data overview

A single toll scheme (and related toll context data) may be split into several parts. Each part of the toll scheme may be of different charging type (i.e. section charging, area charging and cordon charging), may have an individual layout and thus may require different toll context data. This part of ISO 17575 supports this concept by so-called toll context partitions. Details are specified in Clause 8.

The organization of the memory and the physical structure of the data within a Front End are outside the scope of this part of ISO 17575.

6 Procedural requirements and encoding rules

6.1 General

This clause provides normative requirements with regards to

- communication services to be applied for the exchange of context data,
- provisions offered in this part of ISO 17575 in order to enable version and validity control of context data, and
- encoding rules to be applied for context data.

6.2 Communication services

For the purpose of transmitting ADUs from the Back End to the Front End, the communication services defined in ISO 17575-2:2016 or any other appropriate communication services may be used.

NOTE 1 Details with respect to communication services are usually agreed between the operating entities of Back End and Front End.

NOTE 2 State-of-the-art communication frameworks (so-called middleware) designed for data exchange between IT systems and subsystems are appropriate candidate solutions.

6.3 Version and validity handling

6.3.1 Protocol versioning

The Back End shall provide, with each submission of toll context data to the Front End, the application interface definition (syntax and semantics) that is used by the Back End by means of the `protocolVersion`.

The `protocolVersion` information shall be part of the ADU header of the message. The specification of the `protocolVersion` information is provided in [7.4](#) and in [A.2](#).

In cases where the receiving Front End does not support the application interface definition (as indicated by the `protocolVersion`) the Back End requests, the Front End shall provide a negative response to the Back End.

NOTE ISO/TS 17575-3:2011 id not support the version handling of application interface definitions by means of `protocolVersion`.

6.3.2 Context data versioning

Each EFC attribute includes an optional data element containing version and validity information applicable for the respective EFC attribute. The data type of this data element shall always be `VersionAndValidity`. This data type shall comprise two data elements:

- `version`;
- `validFrom`.

The data element `version` shall give the version number of the respective EFC attribute. The data type shall be `VersionId` defined in ISO 17575-1:2016. The version number shall be used in an increasing order.

NOTE 1 This concept enables the Front End to autonomously detect missing versions of context data and potentially initiate an action to update the respective information in the Front End.

The data element `validFrom` shall give the start date and time of the validity of the respective EFC attribute. The data type shall be `GeneralizedTime` as defined in ISO 14906:2011/Amd1:2015.

The information regarding version and validity of EFC attributes enables the Front End to autonomously notice the existence of new updated context data in the Back End.

NOTE 2 Once the start date and time of context data are reached, previous versions (having a version number lower than the current one) become obsolete. The Front End can decide – depending on local settings – to initiate an action to activate the valid context data and deactivate (and delete) the previous used version(s).

The given version and validity information shall be exclusively valid for the EFC attribute it belongs to.

NOTE 3 This concept allows the efficient use of different versions for different types of context data. For example, the tariff table version can be managed independently from the one valid for context layout and reporting rules. This approach reduces the amount of data to be updated.

NOTE 4 The update process itself is outside the scope of this part of ISO 17575.

The optional data element `tollContextVersion` shall be used for the version identification for the entire toll context description. This data element is a component of the data type `ISO17575-3AduBody`. The data type shall be `Int1`.

NOTE 5 If and how such context version identification is composed, using the version information of the underlying attributes and data elements (i.e. vehicle class definitions, layout description, charge reporting events, charge report configuration, tariff definitions), depends on the particular operational requirements and implementation solution.

NOTE 6 One potential use of such context version identification is in relation to compliance check communication (CCC) and the related data exchange. ISO 12813:2015 specifies a context version identifier (also of type `Int1`) that may be provided from the Front End to enforcement facilities. Based on such data, a simple check is possible to identify Front Ends in enforcement processes that use an outdated version of the toll context description.

6.4 Encoding rules

The data types and associated coding related to the data elements described in [Clauses 7](#) and [8](#) are defined using the Abstract Syntax Notation One (ASN.1) technique according to ISO/IEC 8824-1 (see [Annex A](#)).

The encoding rules (e.g. Basic, Packed or XML Encoding Rules, BER, PER or XER) are not specified in this part of ISO 17575 because the physical implementation of the ISO 17575 interface can vary widely. Therefore, the choice of encoding rules shall be adapted to the specific needs (e.g. coding efficiency, compatibility with existing software environments, etc.).

6.5 Acknowledgement and behaviour on errors

The interface between Front End and Back End is located within the realm of the entity acting as toll service provider. Therefore, it can be expected that this interface is not necessarily an interoperable interface.

In order to keep the freedom and flexibility for existing and future implementation options and to not over-specify requirements that are in the sole realm of one designing entity, this part of ISO 17575 does not contain any requirements with regards to acknowledgements and behaviour on errors. It is expected that the entity designing this interface makes the appropriate provisions to ensure a reliable and stable operation of this interface and the transfer of toll context data under any possible conditions.

7 Application data units

7.1 General

This clause provides normative requirements with regards to information used to manage the exchange of toll context data. In order to provide the receiving end with the interface appropriate management information, such data are made available, in addition to the toll context data, in a dedicated protocol header (see [7.3](#)). The header consists of various data elements, as specified in [7.4](#) below. The context

data are considered as the “payload” of the data exchange and are made available in the ADU body (see [7.3](#) and [7.5](#)).

7.2 Message authentication (data type Iso17575-3-InformationContent)

To ensure an uninterrupted chain of trust, security mechanisms are implemented for proof of authenticity and integrity of the transmitted application data units.

To support an integer and secure transfer of toll context data from the Back End to the Front End, optional authentication mechanisms have been introduced and may be used by the involved entities.

Consequently, and depending on the use of the optional authentication, the information content is either an

- unauthenticated application data unit (notAuthenticatedIso17575-3Adu of type Iso17575-3Adu); or
- authenticated application data unit (authenticatedIso17575-3Adu of type AuthenticatedIso17575-3Adu).

The detailed specification of the data type Iso17575-3Adu is provided in [7.3](#).

The data type AuthenticatedIso17575-3Adu comprises the following data elements:

- iso17575-3AduPer;
- messageAuthenticator.

The data element iso17575-3AduPer is a container (of type BIT STRING) that shall hold a data element of type Iso17575-3Adu, which is encoded using ASN.1 Packed Encoding Rules aligned according to ISO/IEC 8825-2.

The data element messageAuthenticator (of type MessageAuthenticator) shall hold the authenticator, which shall be calculated over the single bit string content of iso17575-3AduPer.

NOTE The data type MessageAuthenticator is imported from and defined in ISO 17575-1:2016. Details with respect to functionalities, data elements and options provided in the entire message authentication data structure are described in ISO 17575-1:2016, ISO/TS 19299:2015 and in the underlying data security standards as referenced by ISO 17575-1:2016.

7.3 Application data unit structure (data type Iso17575-3Adu)

For the purpose of context data transfer and context data identification, the message content shall be structured into application data units (ADU), see [Figure 4](#).

Each ADU shall consist of an

- ADU header (of data type Iso17575-3AduHeader), and
- ADU body (of data type Iso17575-3AduBody).



Figure 4 — Structure of an ISO 17575-3 ADU

7.4 Application data unit header (data type ISO 17575-3AduHeader)

The ADU header shall provide management information that applies in relation to the payload (i.e. context data) provided in the ADU body. The Front End may need this information for internal processes of context data management, storage and processing.

The information provided in the ADU header is valid for the entire set of context data provided in the ADU body.

The ADU header shall consist of the following data fields ([Figure 5](#)):

- applied protocol version;
- information sender identification;
- information originator identification;
- toll context identification;
- ADU sequence number;
- message date.

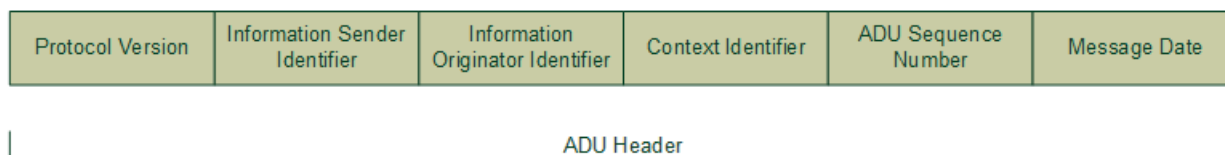


Figure 5 — Structure of the ADU header

The semantics for the data elements of the ADU header shall be according to [Table 1](#). The data types are specified in [Annex A](#).

Table 1 — Data elements of the ADU Header

Data element	Data type (informative)	Definition of semantic	Remarks
protocolVersion	AidIdentifier	Identifier of the version of the ASN.1 data specification (version of ISO 17575) that is used in the application data units	e.g. "1" = ISO17575-3:2016 (see also 6.3.1)
informationSender	Provider	Unique identifier of the entity which has sent the entire message	Usually a service provider operating the Back End, but may also be e.g. a transport service provider, toll charger
informationOriginator	Provider	Unique identifier of the entity which has created the context data provided in the ADU body	e.g. transport service provider, toll charger or service provider
tollContext	Provider	Unique identifier of the entity which acts as toll charger of the toll scheme to which the toll context data belong	Consists of country code and unique number within the country
aduSequenceNumber	Int4	Sequence number of the respective ADU. Shall be used in increasing order. In case of overflow the sequence number shall restart at '0'.	
messageDate	GeneralizedTime	Date and time when the message is generated by the informationSender	Can be used for message security purposes, e.g. for the generation of the authenticator

7.5 Application data unit body (data type ISO 17575-3AduBody)

The ADU body shall contain one or more EFC attributes describing the toll context. One ADU body shall contain EFC attributes belonging to one single toll context only.

NOTE Very complex toll contexts (e.g. containing different types of toll schemes such as a cordon charging and a section charging) can be split into more than one toll context partitions. Interdependencies (priorities) between the individual toll context partitions are specified by means of priority rules.

The toll context for which the data in the ADU body provide detailed description is identified by information given in data element `tollContext` in the ADU header.

EXAMPLE 1 The ADU body contains the tariff table for the Barcelona congestion charging scheme.

EXAMPLE 2 The ADU body contains the geographic description of the road network of the truck tolling schemes applicable on highways in Hungary.

EXAMPLE 3 The ADU body contains the reporting rules applicable in the nationwide all-road charging scheme in Denmark.

EFC attributes that hold toll context data are defined in [Clause 8](#).

The ADU body also contains the optional data element `tollContextVersion`. This data element can be used to specify the version of the entire toll context description (see also [6.3.2](#)).

8 EFC Attributes

8.1 General

This clause provides normative requirements with regards to the EFC attributes holding the context data in terms of their use, structure and semantics.

8.2 Rules with respect to support of context data

Context data available in the Front End shall contain all information required to ensure a minimum level of functionality to either participate in the services of this toll scheme or to unambiguously identify the toll scheme as being not valid e.g. for the specific user, vehicle, moment in time.

Each EFC attribute being part of the context data shall be allocated to one single toll context.

8.3 Attributes and data sets

Each EFC context shall be described using one or more EFC attributes. The full set of EFC attributes belonging to one EFC context shall contain all necessary information to enable proper functioning of the Front End in the respective EFC scheme.

For readability, in this part of ISO 17575 the EFC attributes have been logically structured into data sets.

The following data sets are used:

- Context Overview;
- Tariff Information;
- Context Layout;
- Reporting Rules.

NOTE Logical data sets are fully independent from the physical data structure in a Front End. The physical structure is implementation dependent and outside the scope of this part of ISO 17575.

8.4 EFC attributes authentication

This part of ISO 17575 provides measures that allow for optional authentication of the individual EFC attributes. In contrast to the message authentication (see 7.2) this option can be used for true end-to-end security covering the full chain from the originator of context data to the Front End.

EXAMPLE Such end-to-end security allows a Front End module (which can e.g. reside in an OBE) to check the authenticity, completeness and integrity of a tariff table that is created by the toll charger.

EFC attributes being part of the ADU body can be present in an

- unauthenticated form (data element unsigned-data shall be used), or
- authenticated manner (data element signed-data shall be used).

In cases where the authenticated option is chosen, the data type shall be `Signed{Payload}`, which comprises the following data elements:

- payload;
- `timeOfAuthentication`;
- authenticator.

The data element `payload` is a container (of type `BIT STRING`) that shall hold a data element of type `ISO17575-3Adu`, which is encoded using ASN.1 Packed Encoding Rules aligned according to ISO/IEC 8825-2. The parameter `Payload` is replaced by the data type of the individual EFC attribute.

The data element `timeOfAuthentication` shall hold the information of the point in time the authenticator was calculated.

The data element `authenticator` (of type `MessageAuthenticator`) shall hold the authenticator, which shall be calculated over the single bit string content of `payload`.

The data type `MessageAuthenticator` is imported from and defined in ISO 17575-1:2016. Details with respect to functionalities and options provided in the entire message authentication data structure are described in ISO 17575-1:2016, ISO/TS 19299:2015 (and the underlying data security standards as referenced by ISO 17575-1:2016).

NOTE Not all of the possible options provided by the data structure described in ISO 17575-1:2016 are appropriate for EFC attribute authentication.

8.5 EFC attributes data catalogue

8.5.1 General

The following EFC attributes or a subset here of shall be available to the Front End ([Table 2](#)).

Table 2 — List of EFC attributes

EFC attribute	Remark	Data set	Relevant section
TollContextOverview		Context Overview	8.5.2.2
TollContextPartitionOverview	Defined as list (SEQUENCE OF), which contains one list entry per toll context partition		8.5.2.3
TariffTable	Defined as list (SEQUENCE OF), which contains one list entry per toll context partition	Tariff Information	8.5.3.2
CurrencyConversionTable			8.5.3.2.3
TariffClassDefinition			8.5.3.3
LocalVehicleClassDefinition			8.5.3.4
TimeClassDefinition			8.5.3.5
UserClassDefinition			8.5.3.6
TollContextPartitionLayout	Defined as list (SEQUENCE OF), which contains one list entry per toll context partition	Context Layout	8.5.4
ChargeReportingEvents	Defined as list (SEQUENCE OF), which contains one list entry per toll context partition	Reporting Rules	8.5.5.2
ChargeReportConfiguration	Defined as list (SEQUENCE OF), which contains one list entry per toll context partition		8.5.5.3
TollContextVersion		Version Information	6.3.2

In the following clauses, EFC attributes and data elements are specified in terms of

- names of the data elements forming the EFC attributes,
- content and semantic definition of the EFC attributes and data elements, and
- informative remarks, including references to other standards.

The specification of the corresponding data types in ASN.1 is provided in [Annex A](#).

8.5.2 Requirements with regards to context overview

8.5.2.1 Concept of toll context partitions

One single toll context shall consist of one or more toll context partitions.

Each of the individual partitions shall get an identifier that is unique within the toll context.

Toll context partitions may overlap in terms of their geographical extensions. In cases where they overlap, priority rules ensure the proper functionality according to the applicable toll regulations. The toll charger is responsible for defining the correct priority rules in the toll context data.

8.5.2.2 Toll context overview

The attribute `TollContextOverview` shall contain information about toll context identification, number of present toll context partitions and additional supporting information for the toll context.

The main purpose of the EFC attribute `TollContextOverview` is to give the Front End a minimum amount of basic information regarding a toll scheme. Based on these overview data, the Front End may or may not require and request more information.

EXAMPLE Based on the information in data element `tollContextBoundingBoxes` a Front End dedicated to a passenger car can notice that it is moving in a toll domain. Using the information in the data element `operationalStatus` the Front End algorithms identify this toll scheme is currently inactive. In this case, the Front End does not require additional data (such as context layout or tariff table) of this respective toll scheme.

The structure and data elements of the EFC attribute `TollContextOverview` are given in [Table 3](#) and defined in [Annex A](#).

Table 3 — EFC attribute `TollContextOverview` (informative)

EFC attribute	Data element	Data type	Remark
<code>TollContextOverview</code>	<code>tollContext</code>	Provider	
	<code>tollContextPartitions</code>	SEQUENCE OF <code>TollContextPartitionID</code>	
	<code>tollSchemeName</code>	UTF8String	optional
	<code>tollContextBoundingBoxes</code>	SEQUENCE OF <code>SphericalBox</code>	optional
	<code>TollContextOverviewVersion</code>	VersionAndValidity	

The data element `tollContext` shall identify the toll charger that operates the toll scheme for which the context description is valid. The data type shall be `Provider` as specified in ISO 14906:2011/Amd1:2015.

The data element `tollContextPartitions` shall contain the list of identifiers of the toll context partitions belonging to the single toll context. Each list entry is of data type `TollContextPartitionID`.

At least one toll context partition identifier shall be present in the list. The toll context partition identifiers shall be uniquely defined within the single toll context.

The data element `tollSchemeName` shall contain a designation for the toll scheme. The data element shall be optional. The data type shall be `UTF8String`.

EXAMPLE This data element is used to display a well-known “brand name” of the toll scheme in the OBE (e.g. “LKW Maut”, “Go Maut” or “TIS-PL”).

The data element `tollContextBoundingBoxes` shall contain a list of spherical rectangles that enclose the geographic area of the toll context. Each list entry shall be of data type `SphericalBox`. The data type `SphericalBox` shall contain the description of a spherical rectangle by defining the edges of the rectangle in latitude and longitude. The data element `SphericalBox` shall contain

the data elements `southernLatitude`, `northernLatitude`, `westernLongitude` and `easternLongitude`. The data elements in the data type `SphericalBox` shall be used as follows:

- data element `southernLatitude` shall contain the lower (southernmost) latitude value (data type `Latitude`);
- data element `northernLatitude` shall contain the larger (northernmost) latitude value (data type `Latitude`);
- data element `westernLongitude` shall contain the lower (westernmost) longitude value (data type `Longitude`);
- data element `easternLongitude` shall contain the larger (easternmost) longitude value (data type `Longitude`).

The data element `TollContextOverviewVersion` shall contain version and validity information for the EFC attribute `TollContextOverview`. The data type shall be `VersionAndValidity`. For details see [6.3](#).

8.5.2.3 Toll context partition overview

The attribute `TollContextPartitionOverview` shall contain information about identification, type of the toll scheme (e.g. section based tolling, area charging, cordon charging), time zone information and information regarding the operational status of the respective toll context partition.

The main purpose of the EFC attribute `TollContextPartitionOverview` is to give the Front End a minimum amount of basic information regarding a toll context partition. Based on these overview data, the Front End may or may not require more information.

The structure and data elements of the EFC attribute `TollContextPartitionOverview` are given in [Table 4](#) and defined in [Annex A](#).

Table 4 — EFC attribute `TollContextPartitionOverview` (informative)

EFC attribute	Data element	Data type	Remark
<code>TollContextPartitionOverview</code>	<code>tollContextPartitionID</code>	<code>TollContextPartitionID</code>	
	<code>tollContextPartitionName</code>	<code>UTF8String</code>	optional
	<code>tollContextPartitionType</code>	<code>TollSchemeType</code>	
	<code>operationalStatus</code>	<code>OperationalStatus</code>	
	<code>timeZone</code>	<code>INTEGER (-720..720)</code>	
	<code>dstOffset</code>	<code>INTEGER (-120..120)</code>	optional
	<code>tollContextPartitionBoundingPolygon</code>	<code>Polygon</code>	optional
	<code>sendChargeReportIfEntering</code>	<code>BOOLEAN</code>	optional
	<code>precedenceLevel</code>	<code>Int1</code>	optional
	<code>chargeReportFinalRecipient</code>	<code>Provider</code>	optional
	<code>TollContextPartitionOverviewVersion</code>	<code>VersionAndValidity</code>	

The data element `tollContextPartitionID` shall contain the identifier of the respective toll context partition for which the overview information is provided. The toll context partition ID provided in this field shall correspond to one element in the list of toll context partition IDs in the attribute `tollContextOverview`.

The data element `tollContextPartitionName` shall contain a designation for the toll context partition. The data element shall be optional. The data type shall be `UTF8String`.

NOTE 1 This data element can be used to display a well-known “brand name” of the toll context at the OBE. (e.g. “LKW Maut”, “Go Maut” or “TIS-PL”).

The data element `tollContextPartitionType` shall contain information regarding the type of the toll context partition (road section charging, distance based area charging, time based area charging, cordon charging).

The data element `operationalStatus` shall contain information regarding the period of operation of the toll context partition. This information shall be given by defining date and time when the toll scheme starts or has started operation (in data element `startsOperationAt`) and optionally by defining date and time when the toll scheme will stop operation (in data element `stopsOperationAt`). Both data elements `startsOperationAt` and `stopsOperationAt` shall be of data type `GeneralizedTime`.

The data element `timeZone` shall give the time zone applicable for the toll context partition in relation to Coordinated Universal Time (UTC). The data type is `INTEGER(-720..720)`. The value shall be given in minutes in relation to UTC (-720 ... +720 minutes). In cases where the toll domain of this toll context partition is located on more than one time zone, this toll context partition shall be split into two separate toll context partitions and have two toll context partition data sets.

All absolute time information used in EFC attributes and data elements defined in this part of ISO 17575 (especially in the time class definitions) shall be given in local real time. In cases where daylight saving time (DST) applies, the Front End shall be capable of re-calculating time information according to the applicable legal rules.

To provide supporting information with respect to DST, the optional data element `dstOffset` shall be used to specify the amount in minutes the DST offsets compared to the standard local time. The data type shall be `INTEGER(-120..120)`. Positive values shall be used in cases where the DST is “ahead” of standard local time. Negative values shall be used in cases where the DST is “behind” the standard local time.

EXAMPLE 1 In cases where the standard local time is 12:00h and `dstOffset` = 60, the resulting applicable local time during DST is 13:00h.

EXAMPLE 2 In cases where the standard local time is 12:00h and the `dstOffset` = -45, the resulting applicable local time during DST is 11:15h.

NOTE 2 The time and date from and to DST is effective is not specified in this part of ISO 17575. Particular rules which are compatible to the legal ground are available in several underlying IT systems and frameworks.

The data element `tollContextBoundingPolygon` shall contain a polygon representation that encloses the geographic area of the toll context partition.

The data type `Polygon` is defined as a list of the data type `Point`. The points shall be defined in an order that creates and closes the polygon in a clockwise direction. The segments of the polygon shall be created by connecting each of the points with its successor point. The polygon shall be closed by connecting the last point in the list with the first point. The points shall be defined in a way that the connections between the points do not intersect. The inner side of the polygon (tolled area) is defined as the geographical area that is located at the right-hand side when moving on a polygon segment from its start point to its end point.

The optional data element `sendChargeReportIfEntering` shall be applied in order to require the generation and transmission of a charge report in cases where the Front End is entering the geographical area of the respective toll context partition. The geographical area of the toll context partition is specified in the data element `tollContextBoundingPolygon`. The resulting charge report may also be empty.

NOTE 3 This feature allows the opening or closing of accounts based on the presence of a toll liable vehicle in a given toll context partition.

The optional data element `precedenceLevel` (of type `Int1`) may be used to specify priorities for overlapping toll context partitions. In cases where two or more toll context partitions have got the same preceded level, charge reports shall be generated and transmitted for each individual toll context partition. In cases where the precedence levels of overlapping toll context partitions have different

values, only charge reports for the toll context partition having the higher precedence level shall be generated and transmitted.

A precedence level value of 0 shall be used for the lowest priority and a value of 255 shall be used for the highest priority.

The optional data element `chargeReportFinalRecipient` may be used to provide supplementary information for the management of charge reports in cases where one toll charger operates the toll scheme of another toll charger. In such cases, a dedicated toll context partition shall be used. The data element shall hold the identification of the entity that is the final destination of the respective usage data generated by the Front End. This data element is of the data type `Provider`, which is imported from ISO 14906:2011/Amd1:2015.

NOTE 4 Small toll context partitions (e.g. a bridge) might be embedded within larger toll contexts (e.g. a national toll scheme). In these cases, the toll charger of the large toll context partition assumes the role of toll charger for all external purposes, but internally the final destination of the charge reports is the toll charger of the small toll context partition. This latter operator is identified with the data element `chargeReportFinalRecipient`.

The data element `TollContextPartitionOverviewVersion` shall contain version and validity information for the EFC attribute `TollContextPartitionOverview`. The data type shall be `VersionAndValidity`. For details see [6.3](#).

8.5.3 Requirements with regards to tariff information

8.5.3.1 Concept and overview

The EFC attributes in the data set Tariff Information shall contain all necessary information regarding

- tariff table,
- tariff classes, and
- locally applicable definitions for vehicle classes, time classes and user classes.

All this information is toll scheme specific.

The data set shall comprise the following EFC attributes:

- `TariffTable`
- `CurrencyConversionTable`
- `TariffClassDefinition`
- `LocalVehicleClassDefinition`
- `TimeClassDefinition`
- `UserClassDefinition`

The descriptions and definitions for these EFC attributes are given in [8.5.3.2](#) to [8.5.3.6](#).

The fee calculation algorithm is specified in [8.5.3.7](#).

8.5.3.2 Tariff table

Each toll context description contains one or more tariff tables. One tariff table is valid for one or more toll context partitions. The tariff table comprises the following information:

- toll context partitions the tariff table is applicable for;

- for each tariff class
 - a unique tariff class identifier,
 - the definition of the charge unit,
 - the rounding rule that applies for the charged units used,
 - a basic fee per charge unit,
 - the rounding rule that applies for the fee calculated,
 - the VAT amount per charge unit (optional),
 - the rounding rule that applies for the VAT calculated (optional),
 - an interval scale parameter (optional),
 - an offset fee (optional),
 - a minimum fee (optional),
 - a threshold fee (optional),
 - a maximum fee (optional), and
 - an alternative currency for this tariff class (optional);
- standard currency used in this tariff table;
- type of the fee;
- version and validity information.

The tariff table provides all the tariff data to the Front End that are required to calculate the fee to be paid by a user under given and defined conditions (e.g. vehicle class, time class, etc.).

8.5.3.2.1 EFC attribute `TariffTable`

The tariff table shall be presented in the EFC attribute `TariffTable`.

The structure and data elements of the EFC attribute `TariffTable` are given in [Table 5](#) and defined in [Annex A](#).

Table 5 — EFC attribute `TariffTable` (informative)

EFC attribute	Data element	Data type	Remark
TariffTable	applicablePartitions	SEQUENCE OF <code>TollContextPartitionId</code>	
	tariffs	SEQUENCE OF <code>Tariff</code>	
	standardCurrency	<code>PayUnit</code>	
	typeOfFee	<code>TypeOfFee</code>	optional
	tariffTableVersion	<code>VersionAndValidity</code>	

The data element `applicablePartitions` shall contain a list of the identifiers of those toll context partitions the tariff table is applicable for. Each list entry shall be of type `TollContextPartitionId`.

The data element `tariffs` shall contain a list of tariff related data for each tariff class. Each of the list entries shall be of data type `Tariff`. This data type is described in [8.5.3.2.2](#).

The data element `standardCurrency` shall be used to determine the currency that shall be used. This currency shall be applicable for all data containing any sort of fee information in the tariffs (see

[8.5.3.2.2](#)) that are `basicFeePerChargeUnit`, `offsetFee`, `minFee`, `thresholdFee` and `maxFee`. The data type shall be `PayUnit` as specified in and imported from ISO 14906:2011/Amd1:2015.

In cases where more than one currency is allowed or required in one toll domain (e.g. for particular user groups or because of legal requirements), the alternative currencies are specified in the respective tariff (see [8.5.3.2.2](#)).

The optional data element `typeOfFee` provides information about the nature of the fee (e.g. tax, duty, custom). The data element shall be of type `TypeOfFee`.

NOTE The presence and content of this data element can be irrelevant for the functioning of the Front End, but it can be of major relevance for the Back End and for the interfaces between the toll charger and the service provider.

The data element `tariffTableVersion` shall contain version and validity information for the EFC attribute `TariffTable`. The data type shall be `VersionAndValidity`. For details see [6.3](#).

8.5.3.2.2 Data type Tariff

The data type `Tariff` shall provide the following information per tariff class:

- a tariff class identifier (to unambiguously identify a tariff class within the toll context description);
- the charge unit (e.g. 100 m);
- rounding rule for the calculation of the charge units used (e.g. always rounding up);
- basic fee per charge unit (e.g. 0,03 EUR);
- rounding rules for the calculated fee (e.g. always rounding down);
- optional VAT amount (e.g. 20,3 %);
- optional rounding rules for the calculated VAT (e.g. always using accounting rounding rules);
- optional interval scale parameter;
- optional offset fee;
- optional minimum fee;
- optional threshold fee;
- optional maximum fee;
- optional alternative currency valid for this particular tariff class.

The structure and data elements of the data type `Tariff` are given in [Table 6](#) and defined in [Annex A](#).

Table 6 — Data type *Tariff* (informative)

Data type	Data element	Data type	Remark
Tariff	tariffClass	TariffClassId	
	chargeUnit	ChargeUnit	
	roundingRuleForChargeUnitsUsed	RoundingRule	
	basicFeePerChargeUnit	Int4	
	roundingRuleForFee	RoundingRule	
	vat	Int4	optional
	roundingRuleForVAT	RoundingRule	optional
	intervalScaleParameters	Sequence of IntervalScaleParameter	optional
	offsetFee	Int4	optional
	minFee	Int4	optional
	thresholdFee	Int4	optional
	maxFee	MaxFee	optional
	alternativeCurrency	Currency	optional

The data element *tariffClass* shall identify the tariff class. The data type shall be *TariffClassId*. This data type is defined as *Int4*. The applicable tariff class is defined in attribute *TariffClassDefinition*, which is defined in [8.5.3.2.3](#).

Data element *chargeUnit* shall contain the charge unit valid for the toll scheme. It contains the smallest interval of time period, distance or number of events that are of relevance for the fee calculation. A charge unit shall have one of the following natures:

- a distance (e.g. 100 m or 1 km);
- a time period (e.g. 10 min, 1 h or 7 h);
- a number of events that have occurred (e.g. 1 passage).

The use of charge units depends on the type of charging scheme. The following combinations of charge unit type and toll scheme type shall be possible:

- distance (e.g. metre, km), which shall only be used in section or area charging schemes;
- time (e.g. seconds, minutes, hours), which shall only be used in area charging schemes;
- event (e.g. number of passages), which shall only be used in cordon or section charging schemes.

EXAMPLE 1 A section charging scheme uses as charge unit 1 km, as the fee depends on the length of the used sections. In this example the fee increases every 1 km.

EXAMPLE 2 An area charging scheme uses as charge unit 1 hour, as the fee depends on the duration of stay in the defined area. In this example the fee increases every hour.

EXAMPLE 3 A cordon charging scheme uses as a charge unit 1 event, as the fee depends on the number of passages of the cordon. In this example the fee increases with every single event.

NOTE 1 In time based charging schemes, the charge unit can also be used to define the smallest possible time periods to be used in the fee calculation.

EXAMPLE 4 A time based charging scheme with a charge unit value of 7 hours is defined. In cases where the basic fee is defined by 5 GBP, a stay in the charging area of less than 7 hours would, in any case, result in a fee of 5 GBP.

The data element *chargeUnit* shall be of data type *ChargeUnit*, which is a CHOICE type and has the type distance, time or event. The data type of the data element distance shall be *Distance*.

as defined in ISO 17575-1:2016. The data type of data element `time` shall be `Duration`, as defined in ISO 17575-1:2016. The data type of the data element `event` shall be `Int1`.

The data element `roundingRuleForChargeUnitsUsed` shall be used to specify if and how rounding of distance driven or time in the toll liable network shall applied when determining the charge units used. The data type shall be `RoundingRules`. This data type is specified as `Int1`. The following values are defined:

- `no (0)` – no rounding is applied;
- `up (1)` – always rounding up to the next full larger value of the charge unit specified in `chargeUnit`;
- `down (2)` – always rounding down to the next full lower value of the charge unit specified in `chargeUnit`;
- `accounting (3)` – always rounding according to methods used in accounting (e.g. as defined in DIN 1333[6]).

EXAMPLE 5 Charge units are specified with 200 m (section based charging). The vehicle has used the toll liable road network for 750 m. A rounding rule is set up to 1 = rounding up. The resulting number of charge units used is 4.

EXAMPLE 6 Charge units are specified with 4 h (area based charging). The vehicle has used the toll liable area for 5:56 h. A rounding rule is set up to 2 = rounding down. The resulting number of charge units used is 1.

The data element `basicFeePerChargeUnit` shall contain the basic fee per charge unit. The data type shall be `Int4`. The value shall be given in the currency and unit as defined in the data element `standardCurrency` in the data element `tariffTable`. The value in the data element `basicFeePerChargeUnit` shall be given excluding VAT.

EXAMPLE 7 A `basicFeePerChargeUnit` value of 175, currency value of “EUR” and a currency unit of “minor units of 1000:1” (see ISO 14906:2011/Amd1:2015) is specified. The resulting fee is 0,175 EUR = 17,5 cent.

The data element `roundingRuleForFee` shall be used to specify if and how rounding of resulting toll fee shall applied when calculating the fee per used charge object (see also 8.5.3.7). The data type shall be `RoundingRules`. This data type is specified as `Int1`. The following values are defined:

- `no (0)` – no rounding is applied;
- `up (1)` – always rounding up to the next larger value of the currencies minor unit;
- `down (2)` – always rounding down to the next lower value of the currencies minor unit;
- `accounting (3)` – always rounding according to methods used in accounting (e.g. as defined in DIN 1333[6]).

The optional data element `vat` shall contain the VAT per charge unit. The data type shall be `Int4`. The value shall be given in 0,01 %.

The optional data element `roundingRuleForVat` shall be used to specify if and how rounding of the resulting VAT for the toll fee shall applied when calculating the VAT per used charge object (see also 8.5.3.7). The data type shall be `RoundingRules`. This data type is specified as `Int1`. The following values are defined:

- `no (0)` – no rounding is applied;
- `up (1)` – always rounding up to the next larger value of the currencies minor unit;
- `down (2)` – always rounding down to the next lower value of the currencies minor unit;
- `accounting (3)` – always rounding according to methods used in accounting (e.g. as defined in DIN 1333[6]).

In cases where the optional data element `vat` is present but the optional data element `roundingRuleForVat` is not present, no rounding shall be applied for the VAT.

The optional data element `intervalScaleParameters` shall contain a list of additional parameters to adapt the basic fee depending on the parameters maximum laden weight of a vehicle and the length of the vehicle. These parameters shall contain offset values (data element `zeroOffset`), resolution data (data element `resolution`) and maximum values (data element `max`).

NOTE 2 Maximum laden weight of vehicles as interval scale parameters are currently in use in the Swiss LSVA toll scheme. The maximum length of a vehicle is also considered useful for payment for use of ferries.

The optional data element `offsetFee` shall be used to specify an offset value that should be added to the calculated fee (see also [8.5.3.7](#)). The data type shall be `Int4`. The value shall be given in the currency and currency units that are specified by the data element `standardCurrency` in the data element `tariffTable`.

The optional data element `minFee` shall be used to specify a minimum fee. This minimum fee may be lower than the fee that would apply according to the real usage of the charged network. The data type shall be `Int4`. The value shall be given in the currency and currency units that are specified by the data element `standardCurrency` in the data element `tariffTable`.

EXAMPLE 8 A minimum fee of 1,00 EUR is defined in a section based charging scheme. However, there are very short sections for which, based on the applicable toll section length and the tariff, a fee of less than 1,00 EUR would apply (e.g. 1 km x 0,10 EUR/km = 0,10 EUR).

The optional data element `thresholdFee` shall be used to specify a threshold fee. In cases where the fee resulting from the real usage of the charged network is lower than the threshold value, no fee shall apply. The data type shall be `Int4`. The value shall be given in the currency and currency units that are specified by the data element `standardCurrency` in the data element `tariffTable`.

EXAMPLE 9 A vehicle uses a time based area charging scheme for 10 minutes. The fee valid for the applicable tariff class is specified by 0,02 EUR per minute. The threshold fee is specified by 0,30 EUR. As the calculated fee does not reach the threshold value, the resulting fee is 0,00 EUR.

The optional data element `maxFee` shall contain a maximum value for the fee valid for the particular tariff class. The data type shall be `MaxFee`.

The data type `MaxFee` allows the specification of different maximum fee values:

- per day;
- per week;
- per month;
- per year.

The value shall be given in data type `Int4` and in the currency and currency units that are specified by the data element `standardCurrency` in the data element `tariffTable`.

NOTE 3 These parameters may be used, for example, in cordon charging schemes to cap the maximum fee due per day in cases where there are multiple passages across the cordon border.

EXAMPLE 10 A `maxFee` value of “11650” and `currency` value of “EUR” specifies 116,50 EUR maximum fee.

The optional data element `alternativeCurrency` shall be used to specify an alternative currency in cases where the toll context uses more than one currency (e.g. for particular user groups or because of legal reasons). If this data element is present for this particular tariff class (identified by a tariff class ID), the alternative currency specified in the data element `alternativeCurrency` shall be valid instead of the standard currency that is specified in the attribute `TariffTable`. However, the data provided in the elements `basicFeePerChargeUnit`, `offsetFee`, `thresholdFee`, `minFee` and `maxFee` shall still be given in the currency and units specified in the tariff table. In these cases, a conversion rate shall be applied between the two currencies. The conversion rates for alternative currencies are provided in the attribute `CurrencyConversionTable` (see [8.5.3.2.3](#)). The data type of the data element `alternativeCurrency` shall be `Currency`. The data type `Currency` provides the currency identification as specified in ISO 4217.

8.5.3.2.3 EFC attribute CurrencyConversionTable

The tariff conversion table shall be presented in the EFC attribute CurrencyConversionTable.

The structure and data elements of the EFC attribute CurrencyConversionTable are given in [Table 7](#) and defined in [Annex A](#).

Table 7 — EFC attribute CurrencyConversionTable (informative)

EFC attribute	Data element	Data type	Remark
CurrencyConversionTable	conversions	Sequence of	
	alternativeCurrency	Currency	
	conversionRate	Int4	
	CurrencyConversionTableVersion	VersionAndValidity	

For each toll domain a standard currency shall be defined in the Tariff Table (data element standardCurrency as specified in [8.5.3.2](#)).

The attribute CurrencyConversionTable shall contain a list of conversion rates that are applicable in cases where the toll context applies more than one currency for the calculation of the fees due.

NOTE Additional currencies can apply for particular user groups or for legal reasons.

Alternative currencies shall be specified individually per tariff class (see [8.5.3.2.2](#)) for the instances to which they apply. In cases where an alternative currency is specified in data element Tariff, the attribute CurrencyConversionTable shall be present.

The currency conversion table contains, for each alternative currency, the identification of the alternative currency and a conversion rate to the standard currency.

The element alternativeCurrency identifies the individual currency by its currency code. The data type shall be Currency, which provides the coding as specified in ISO 4217.

The data element conversionRate provides the conversion rate between the respective alternative currency and the standard currency. The data type shall be Int4. The value shall be given in 0,0001 major units of the alternative currency compared to one major unit of the standard currency.

The following formula shall apply:

$$Value_in_alternative_currency = Value_in_standard_currency \times conversion_rate$$

EXAMPLE 1 The standard currency is specified as EUR. The alternative currency is specified as NOK. The exchange rate assumed 7,6771 NOK = 1 EUR. In this example, the value in the data element conversionRate shall be 76771.

EXAMPLE 2 The standard currency is specified as EUR. The alternative currency is specified as HUF. A basic fee per charge unit for the particular tariff class is specified as 0,11 EUR. The value in the element conversion rate is set to 2948500. The calculated basic fee per charge unit in the alternative currency is 32,4335 HUF (0,11 EUR x 2948500 x 0,0001).

The data element CurrencyConversionTableVersion shall contain version and validity information for the EFC attribute CurrencyConversionTable. The data type shall be VersionAndValidity. For details see [6.3](#).

8.5.3.3 Tariff class definition

8.5.3.3.1 General principle

Tariff classes are always considered to be toll scheme specific. Tariff classes depend on one or more of the following determinants:

- vehicle class;
- time class;
- location class;
- user class.

The four tariff determinants can be defined and used completely independent from each other (orthogonal).

Each valid combination of the tariff determinants vehicle class, time class, location class and user class shall lead to one tariff class (see [Figure 6](#)).

NOTE A “combination” means a logical AND of the four tariff determinants.

However, in cases where a combination of valid time class, valid local vehicle class, valid user class and valid location class occurs during runtime to which no resulting tariff class is specified in the tariff table, the respective Front End shall apply a default vehicle class in the respective charge report(s). The tariff class ID of the default tariff class shall be set to 0. Additional specifications in such cases (e.g. default fees) may be agreed between the toll charger and toll service provider(s).

Different combinations of tariff determinants may lead to the same tariff class.

EXAMPLE Both combinations “time class 1; vehicle class 4; user class 34; location class 21” and “time class 4; vehicle class 4; user class 34; location class 23” lead to tariff class 345.

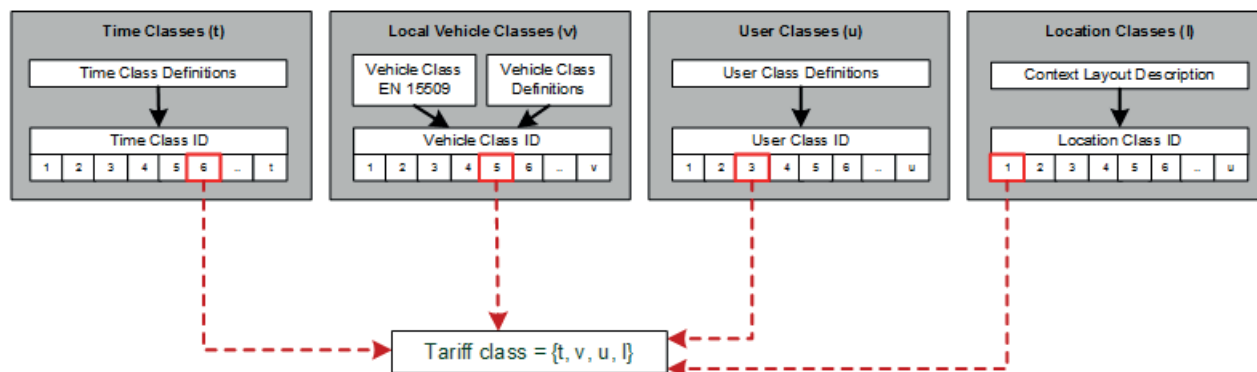


Figure 6 — Principle of creation of tariff classes

There shall be only one applicable tariff class for a user with one single contract for one single vehicle (with a given and fixed characteristic) as long as the vehicle uses one single charge object (with one single location class) at a specific point in time.

8.5.3.3.2 EFC attribute TariffClassDefinition

Tariff classes shall be defined in the EFC attribute `TariffClassDefinition`.

The structure and data elements of the EFC attribute `TariffClassDefinition` are given in [Table 8](#) and defined in [Annex A](#).

Table 8 — EFC attribute `TariffClassDefinition` (informative)

EFC attribute	Data element	Data type	Remark
TariffClassDefinition	tariffClasses	SEQUENCE OF <code>TariffClass</code>	
	tariffClassDefinitionVersion	VersionAndValidity	

The data element `tariffClasses` shall contain a list of data types `TariffClass`. This data type is described in [8.5.3.3.3](#).

The data element `tariffClassDefinitionVersion` shall contain version and validity information for the EFC attribute `TariffClassDefinition`. The data type shall be `VersionAndValidity`. For details see [6.3](#).

This part of ISO 17575 supports different versioning concepts for tariffs, tariff classes, vehicle classes, time classes, user classes and location classes. Changes in tariffs and/or class definitions that apply in the operational lifetime of toll schemes can, for example, be applied the following manner (this is a non-exhaustive list):

- applying the same tariff / class ID after a change in tariffs or class definitions, but using the version and validity data elements in order to identify and specify respective changes in tariffs and classes;
- applying a new tariff / class ID that was not used before the change in tariffs or class definitions and specifying the tariff classes in a way that they comprise only those vehicle / location / time / user classes effective after the change in tariffs or class definitions.

Depending on the method applied in the respective toll context, the optional version and validity information in several class definitions can be present or absent.

8.5.3.3.3 Data type `TariffClass`

The data type `TariffClass` shall contain all valid combinations of the tariff determinants local vehicle class, time class, location class and user class for a tariff class. Tariff classes and the definition of the tariff determinants are applicable for the entire toll context including all toll context partitions.

The structure and data elements of the data type `TariffClass` are given in [Table 9](#) and defined in [Annex A](#).

Table 9 — Data type `TariffClass` (informative)

Data type	Data element	Data type	Remark
TariffClass	tariffClassId	TariffClassId	
	localVehicleClasses	SEQUENCE OF LocalVehicleClassId	
	timeClasses	SEQUENCE OF TimeClassId	optional
	locationClasses	SEQUENCE OF LocationClassId	optional
	userClasses	SEQUENCE OF UserClassId	optional

Each tariff class shall have a unique identifier (data element `tariffClassId` of data type `TariffClassId`).

The data element `localVehicleClasses` shall contain one or more locally defined vehicle classes. The data type shall be `LocalVehicleClassId`. Vehicle classes shall be defined in EFC attribute `LocalVehicleClassDefinition` (see [8.5.3.4](#)).

The optional data element `timeClasses` shall contain one or more time classes. The data type shall be `TimeClassId`. Time classes shall be defined in EFC attribute `TimeClassDefinition` (see [8.5.3.5](#)).

In cases where at least one time class is defined for the toll context, the optional data element `timeClasses` shall be present.

The optional data element `locationClasses` shall contain one or more location classes. The data type shall be `LocationClassId`. Location class information shall be part of the context layout description (see [8.5.4](#)).

In cases where at least one location class is defined for the toll context, the optional data element `locationClasses` shall be present.

The optional data element `userClasses` shall contain one or more user classes. The data type shall be `UserClassId`. User classes shall be defined in EFC attribute `UserClassDefinition` (see [8.5.3.6](#)).

In cases where at least one user class is defined for the toll context, the optional data element `userClasses` shall be present.

In cases where one of the optional elements `timeClasses`, `locationClasses`, `userClasses` is not present, the respective tariff class shall be valid for all possible values of this element.

EXAMPLE 1 Tariff Class 56 is defined as the combination of Local Vehicle Class 23 and User Class 2.

EXAMPLE 2 Tariff Class 106 is defined as the combination of Local Vehicle Classes 23, 78 and 98 Time Classes 32 and 206.

EXAMPLE 3 Tariff Class 87 is defined as the User Class 1.

EXAMPLE 4 Tariff Class 202 is defined as the combination of Local Vehicle Classes 156, 223 and 224 and Time Class 45 and Location Classes 2 and 5 and User Class 2.

8.5.3.4 EFC attribute `LocalVehicleClassDefinition`

8.5.3.4.1 Introduction

The attribute `LocalVehicleClassDefinition` shall contain all necessary data to define vehicle classes that locally apply in the respective toll scheme.

The structure and data elements of the EFC attribute are given in [Table 10](#) and defined in [Annex A](#).

Table 10 — EFC attribute `LocalVehicleClassDefinition` (informative)

EFC attribute	Data element	Data type	Remark
<code>LocalVehicleClassDefinition</code>	<code>localVehicleClasses</code>	SEQUENCE OF <code>LocalVehicleClass</code>	
	<code>localVehicleClassDefinitionVersion</code>	<code>VersionAndValidity</code>	optional

The data element `localVehicleClasses` shall contain a list of data types `LocalVehicleClass`. This data type is described in [8.5.3.4.2](#).

The optional data element `localVehicleClassDefinitionVersion` shall contain version and validity information for the EFC attribute `LocalVehicleClassDefinition`. The data type shall be `VersionAndValidity`. For details see [6.3](#).

8.5.3.4.2 Data type `LocalVehicleClass`

The data type `LocalVehicleClass` contains the definition for one vehicle class. The structure and data elements of the data type `LocalVehicleClass` are given in [Table 11](#) and defined in [Annex A](#).

Table 11 — Data type `LocalVehicleClass` (informative)

Data type	Data element	Data type	Remark
<code>LocalVehicleClass</code>	<code>localVehicleClassId</code>	<code>LocalVehicleClassId</code>	
	<code>nominalElements</code>	<code>NominalVehicleParameters</code>	
	<code>ordinalElements</code>	<code>OrdinalVehicleParameters</code>	optional
	<code>priorityValue</code>	<code>Int1</code>	optional

Each local vehicle class shall have a unique identifier (`localVehicleClassId`) of the data type `LocalVehicleClassId`.

EXAMPLE 1 Vehicle class 21 is defined as all trucks between 3,5 and 12 t and having 3 axles.

EXAMPLE 2 Vehicle class 107 is defined as passenger cars having 3 or 4 axles and euro classes better than 5.

EXAMPLE 3 To exempt vehicles with very low emissions from charges, e.g. in a city charging scheme, these vehicles may be grouped into one vehicle class (e.g. 67), which is defined by all vehicles with euro class 6.

In the context of tariffs, vehicle specific parameters may have a different nature. Vehicle specific parameters may be used as

- nominal scale parameter, or
- ordinal scale parameter.

The data elements `nominalElements` and `ordinalElements` (optional) shall contain the description of vehicle parameters for each local vehicle class.

Each locally defined vehicle class may depend on one or more vehicle specific parameters. Vehicle specific parameters are already defined as attributes in ISO 14906:2011/Amd1:2015 for the purposes of EFC.

NOTE 1 This part of ISO 17575 uses existing data elements (defined in ISO 14906:2011/Amd1:2015) wherever useful.

NOTE 2 Depending on the toll scheme, some of the vehicle parameters can be used in a different manner, e.g. as ordinal scale parameter in one toll scheme and as nominal scale parameter in a second toll scheme.

The data element `nominalElements` shall have the data type `NominalVehicleParameters`. This data type is defined in [8.5.3.4.3](#).

The data element `ordinalElements` shall have the data type `OrdinalVehicleParameters`. This data type is defined in [8.5.3.4.3](#).

Each vehicle class may contain priority information (`priorityValue`) of data type `Int1`. This data element shall be used to determine which vehicle class is valid in cases where more than one vehicle class is applicable (e.g. due to overlapping class definitions). A value of 0 shall be used for the lowest priority level and a value of 255 shall be used for the highest priority level.

EXAMPLE 4 Vehicle class 45 is defined as all trucks having an overall length of 6,50 m to 12,50 m. Priority level for this class is set to 8. Vehicle class 46 is defined as all trucks having an overall length of 8,00 m to 15,00 m. Priority level for this class is set to 5. In cases where a vehicle would fall into both classes, as it has a length of 10,65 m, it has to be grouped into vehicle class 45 as the priority level of class 45 is higher.

In cases where vehicle classes overlap (one single vehicle falling into more than one vehicle class), the data element `priorityValue` shall be present. In these cases, the priority values of (partly) overlapping vehicle classes shall have different values.

8.5.3.4.3 Data type NominalVehicleParameters

The data type `NominalElements` shall contain a list of all applicable values of one or more of the following predefined data types:

- `VehicleClass` (as defined in EN 15509:2014);
- `VehicleTrainAxles` (same use as defined in ISO 14906:2011/Amd1:2015 for `TractorAxles` and `TrailerAxles`);
- `EuroValue` (as defined in ISO 14906:2011/Amd1:2015);
- `CopValue` (as defined in ISO 14906:2011/Amd1:2015).

EXAMPLE Vehicle class 31 is defined as all vehicles showing a combination of a `VehicleClass` content “Trucks above 12t”, a `VehicleAxles` content of “2 axles” or “3 axles” and a `EuroClass` content of “Euro3”.

Data content, semantics and format of the data types `VehicleClass`, `VehicleTrainAxles`, `EuroValue` and `CopValue` shall be according to their definition in EN 15509:2014 and ISO 14906:2011/Amd1:2015, respectively.

8.5.3.4.4 Data type OrdinalVehicleParameters

The optional data type `OrdinalVehicleParameters` shall contain a list of the applicable value ranges of none, one or more of the following predefined optional data types, all defined in ISO 14906:2011/Amd1:2015 except where indicated:

- `VehicleLengthOverall`
- `VehicleHeightOverall`
- `VehicleWidthOverall`
- `VehicleFirstAxlesHeight`
- `VehicleTractorAxlesNumber`
- `VehicleTrailerAxlesNumber`
- `VehicleMaxLadenWeight`
- `VehicleTrainMaximumWeight`
- `VehicleWeightUnladen`
- `VehicleWeightLaden`
- `EuroValue`
- `CopValue`
- `VehicleClass` (defined in EN 15509:2014)
- `CO2EmissionValue`
- `DieselEmissionValue`
- `ExhaustEmissionValues`

Data content, semantics and format (including resolution if applicable) of the data types above shall be according to their definition in ISO 14906:2011/Amd1:2015 (or in EN 15509:2014 for `VehicleClass`).

The data element `lowerLimit` shall contain the lower limit of the respective parameter range. The value of the lower limit shall be included in the range.

The data element `upperLimit` shall contain the upper limit of the respective parameter range. The value of the upper limit shall not be included in the range.

EXAMPLE 1 A range defined with `lowerRange` = 3,5 t and `upperRange` = 12 t includes vehicles having 7,5 t in the respective vehicle class.

EXAMPLE 2 A range defined with `lowerRange` = 3,5 t and `upperRange` = 12 t includes vehicles having exactly 3,5 t in the respective vehicle class. It excludes vehicles having exactly 12 t from the respective vehicle class.

8.5.3.5 EFC attribute `TimeClassDefinition`

8.5.3.5.1 Introduction

The attribute `TimeClassDefinition` shall contain necessary definitions for all time classes which apply in the respective toll scheme.

The structure and data elements of the EFC attribute are given in [Table 12](#) and defined in [Annex A](#).

Table 12 — EFC attribute `TimeClassDefinition` (informative)

EFC attribute	Data element	Data type	Remark
TimeClassDefinition	timeClasses	SEQUENCE OF TimeClass	
	timeClassDefinitionVersion	VersionAndValidity	optional

The data element `timeClasses` shall contain a list of data types `TimeClass`. This data type is described in [8.5.3.5.2](#).

The optional data element `timeClassDefinitionVersion` shall contain version and validity information for the EFC attribute `TimeClassDefinition`. The data type shall be `VersionAndValidity`. For details see [6.3](#).

8.5.3.5.2 Date type `TimeClass`

The data type `TimeClass` contains the definitions for one time class. The structure and data elements of the data type `TimeClass` are given in [Table 13](#) and defined in [Annex A](#).

Table 13 — Data type `TimeClass` (informative)

Data type	Data element	Data type	Remark
TimeClass	timeClassId	TimeClassId	
	nominalElements	NominalTimeParameters	optional
	ordinalElements	SEQUENCE OF OrdinalTime-Parameters	optional
	priorityValue	Int1	optional

Each time class shall have an identifier (`timeClassId`) of the data type `TimeClassId`.

The data elements `nominalElements` and `ordinalElements` shall contain the definitions for each time class.

All absolute time information in these data elements shall be given in local real time.

Time class definitions may be based on

- nominal scale parameter, and
- ordinal scale parameter

For each defined time class, at least one type of parameter – nominal scale or ordinal scale – shall be specified. In cases where both nominal and ordinal scale parameters are present in the definition of an individual time class, a Boolean OR combination shall apply.

If, at a certain point in time, the conditions given in nominal parameters, the definitions given in ordinal parameters or the definitions in both parameters are met, then the respective time class shall be active.

Nominal scale parameters shall be defined in data element `nominalElements`. The data type is `NominalTimeParameters`. Nominal scale parameters shall be defined in one or more of the following data elements:

- `weekdays`
- `dates`
- `externalSetClasses`

The data element `weekdays` shall be used to define one or more days of the week (e.g. Monday and Friday). Data type is `Weekday`, which is of data type `Int1`.

EXAMPLE 1 All odd weekdays of the year are coded using the nominal time parameters. This means that in data type `NominalTimeParameters` the element `weekdays` is present and contains the content (1), (3), (5), (7). The optional data elements `dates` and `classesSetExternally` are not present

The data element `dates` shall be used to identify one or more particular dates (e.g. 25.12.2017). Data type is `DateCompact`, as defined in ISO 14906:2011/Amd1:2015.

The optional data element `classesSetExternally` shall be used to set any predefined time class by external input sources independently from the effective time class.

NOTE 1 This function can be used to overrule the time class (and tariff), which would apply based on other time class determinants and the local time. This function can be used in cases where there is a fully dynamic congestion charging scheme. In such schemes, time classes (and tariffs) can be selected based on the real time traffic situation. The external source can, for example, be the Back End or a dedicated short-range communication (DSRC) device installed at the road side.

EXAMPLE 2 A toll charger wants to run a toll scheme in which a particular expensive tariff is valid during periods of congestion. However, the toll charger does not want to predefine the period of congestion as a fixed period, but to apply this tariff only when real congestion is present. In periods of normal traffic, the predefined tariff (based on the predefined time class) is valid. If suddenly the traffic amount exceeds a certain limit, then the toll charger increases the tariff to make use of the roads less attractive and keep the traffic demand at a lower level. In these cases, the special congestion tariff is applied by activating a new time class (`classesSetExternally`) that has higher priority compared to the predefined time class.

In cases where one instance of the data element `nominalTimeParameters` contains more than one entry (in one or more of the data elements `weekdays`, `dates` and `externalSetClasses`), this shall be understood as Boolean OR combination.

Ordinal scale parameters shall be defined in data element `ordinalElements`, which is defined as a SEQUENCE OF the data type `OrdinalTimeParameters`. Ordinal scale parameters shall be defined in one or more of the following data elements:

- `weekdays`
- `absoluteTimesOfDay`
- `relativeTimePeriods`
- `periodsInYear`