
**Electronic fee collection — Evaluation
of equipment for conformity to ISO
17575-1 —**

**Part 1:
Test suite structure and test purposes**

*Perception du télépéage — Évaluation de la conformité de
l'équipement à l'ISO 17575-1 —*

Partie 1: Structure de la suite d'essais et objectifs des essais



STANDARDSISO.COM : Click to view the full PDF of ISO 16407-1:2017



COPYRIGHT PROTECTED DOCUMENT

© ISO 2017, Published in Switzerland

All rights reserved. Unless otherwise specified, no part of this publication may be reproduced or utilized otherwise in any form or by any means, electronic or mechanical, including photocopying, or posting on the internet or an intranet, without prior written permission. Permission can be requested from either ISO at the address below or ISO's member body in the country of the requester.

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

Contents

	Page
Foreword	iv
Introduction	v
1 Scope	1
2 Normative references	2
3 Terms and definitions	2
4 Abbreviated terms	4
5 Test suite structure	5
5.1 Structure	5
5.2 Reference to conformance test specifications	5
5.3 Test purposes (TP)	6
5.3.1 TP definition conventions	6
5.3.2 TP naming conventions	6
5.4 Conformance test report	7
Annex A (normative) Test purposes (TP) for Front End	8
Annex B (normative) Test purposes (TP) for Back End	93
Annex C (normative) Data structures	97
Annex D (normative) PCTR for Front End	99
Annex E (normative) PCTR for Back End	104
Bibliography	108

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

This document was prepared by Technical Committee ISO/TC 204, *Intelligent transport systems*.

This first edition cancels and replaces ISO/TS 16407-1:2011, which has been technically revised.

The main changes compared to the previous edition are as follows:

- this document has been converted from a Technical Specification to an International Standard;
- this document has been amended to reflect changes to the underlying base standards, especially ISO 17575;
- this document contains major changes regarding
 - data element changes introduced by ISO 17575-1 and ISO 17575-3,
 - new test purposes related to protocol version handling and authenticated data elements,
 - removed test purposes related to rules with respect to charging which are not anymore required by ISO 17575-1,
 - revised terms and definitions, and
 - editorial and formal corrections, as well as changes to improve readability.

A list of all parts in the ISO 16407 series can be found on the ISO website.

Introduction

This document, is part of a series of International Standards that supports interoperability of autonomous electronic fee collection (EFC) systems. Autonomous systems use satellite positioning, often combined with additional sensor technologies, such as gyroscopes, odometers and accelerometers, to localize the vehicle and to find its position on a map containing the charged geographic objects, such as charged roads or charged areas. From the charged objects, the vehicle characteristics, the time of day and other data that are relevant for describing road use, the tariff and ultimately, the road usage fee is determined.

Autonomous on-board equipment (OBE) operates without relying on dedicated road-side infrastructure by employing wide-area technologies such as Global Navigation Satellite Systems (GNSS) and Cellular Communications Networks (CN). Therefore, autonomous systems can also be referred to as GNSS/CN systems.

Within the series of EFC standards, this document defines tests for conformity evaluation of Front End and Back End that comply with the requirements towards the charging specified in ISO 17575-1.

This document is based on ISO 17575-1 and the ISO/IEC 9646 series of standards on conformance test methodology.

STANDARDSISO.COM : Click to view the full PDF of ISO 16407-1:2017

Electronic fee collection — Evaluation of equipment for conformity to ISO 17575-1 —

Part 1: Test suite structure and test purposes

1 Scope

The ISO 16407 series of standards specifies a suite of tests in order to assess the Front End and Back End behaviour compliancy towards the requirements listed in ISO 17575-1. This document contains the definition of such tests in the form of test purposes, listing the required initial conditions, references and individual steps in a structured textual manner.

Test purposes defined in this document reflect the structural and semantical requirements stated in ISO 17575-1:

- presence/absence of particular data elements;
- semantics related to various data elements:
 - data group General (see ISO 17575-1:2016, 7.3);
 - data group Security (see ISO 17575-1:2016, 7.4);
 - data group Contract (see ISO 17575-1:2016, 7.5);
 - data group Usage (see ISO 17575-1:2016, 7.6);
 - data group Account (see ISO 17575-1:2016, 7.7);
 - data group Versioning (see ISO 17575-1:2016, 7.8).

With regard to the individual data sets and EFC attributes defined in ISO 17575-1, the test purposes have been organized into the test suite groups designated for the Front End and Back End, respectively.

Besides the test purposes, this document also specifies proforma conformance test report templates for both the Front End and Back End test purposes.

For more information regarding the requirements against which the conformance is evaluated in this document, see ISO 17575-1.

Testing of the following behaviours and functionalities is outside of the scope of this document:

- dynamic behaviour, i.e. sequence of messages and triggering events that can be exchanged/happen to fulfil certain charging scenarios;
- profiles and business logic built on top of particular pricing schemas;
- as ISO 17575-1 does not specify any Behaviour Invalid of Front End and Back End, BI test purposes are not applicable for any test purpose group.

2 Normative references

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

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

3 Terms and definitions

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

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

- IEC Electropedia: available at <http://www.electropedia.org/>
- ISO Online browsing platform: available at <http://www.iso.org/obp>

- 3.1**
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* (3.9) 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

[SOURCE: ISO 17575-1:2016, 3.10, modified — Note 1 to entry has been deleted]

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, modified — The term “road” has been added in the definition as well as Note 1 to entry.]

Note 1 to entry: The Front End comprises the OBE and an optional proxy.

3.12**layout**

technical description of the location of tolled objects including their borders

[SOURCE: ISO 17575-3:2016, 3.12]

3.13**on-board equipment****OBE**

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

[SOURCE: ISO 17575-3:2016, 3.13]

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 to using a vehicle in a *toll domain* (3.20)

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.

[SOURCE: ISO/TS 19299:2015, 3.42, modified — The phrase “in relation with” has been changed to “in connection to” in the definition.]

3.17

tolled area

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

[SOURCE: ISO 17575-3:2016, 3.17]

3.18

toll context

logical view as defined by *attributes* (3.2) and functions of the basic elements of a *toll scheme* (3.22) 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 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]

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

[SOURCE: ISO 17575-3:2016, 3.22]

4 Abbreviated terms

ADU	application data unit (see ISO 14906)
ASN.1	Abstract Syntax Notation One (see ISO/IEC 8824-1)
ATS	abstract test suite
BI	behaviour invalid
BV	behaviour valid (see EN 15876-1)
CCC	compliance check communication (see ISO 12813)
CN	cellular network
DUT	device under test

EFC	electronic fee collection (see ISO 14906)
GNSS	Global Navigation Satellite Systems
HMI	human machine interface
ID	identifier
IUT	implementation under test
OBE	on-board equipment (see ISO 14906)
PCTR	proforma conformance test report
PICS	protocol implementation conformance statements
TP	test purposes
TSS	test suite structure
TTCN	testing and test control notation
VAT	value added tax

5 Test suite structure

5.1 Structure

[Table 1](#) shows the test suite structure (TSS).

Table 1— Test suite structure

Group	Type of IUT	Behaviour
Charge report	Front End	Behaviour valid
		Behaviour invalid not applicable
Back end feedback	Front End	Behaviour valid
		Behaviour invalid not applicable
Charge report response	Back End	Behaviour valid
		Behaviour invalid not applicable

5.2 Reference to conformance test specifications

This document takes into account already defined test purposes for conformance to the base standards by referencing them, so that

- for test purposes that are identical to those defined in this document or the base standards conformance test cases direct reference is reported; for reader's convenience, the title or a verbal description of the referenced test purpose is given, together with the reference,
- for test purposes that are derived from those defined in the base standards conformance test cases, a direct reference is reported, plus an indication on how the referred test purpose shall be modified for the profile conformance testing,
- for test purposes that are specific to ISO 17575-1, a complete description is given, and
- an indication on whether a test purpose is identical, derived, or specific is given in each test purpose.

5.3 Test purposes (TP)

5.3.1 TP definition conventions

The TPs are defined following the rules shown in [Table 2](#). All test purposes are defined in [Annex A](#) and [Annex B](#), including the special notation and symbol conventions that shall be used. The data structures that shall be used are specified in [Annex C](#) and defined in ISO 17575-1.

Table 2 — TP definition rules

TP ID according to the TP naming conventions	Title
	Reference
	TP origin
	Initial condition
	Stimulus and expected behaviour

TP ID	The TP ID is a unique identifier. It shall be specified according to the TP naming conventions defined in 5.3.2 .
Title	Short description of test purpose objective.
Reference	The reference should contain the references of the subject to be validated by the actual TP (specification reference, clause, paragraph) or the reference to the standard document defining the TP.
TP origin	Indicates if the TP is identical to a TP defined in another test standard, derived from a TP defined in another test standard or specific for this standard profile.
Initial condition	The condition defines in which initial state the IUT has to be to apply the actual TP.
Stimulus and expected behaviour	Definition of the events the tester performs and the events that are expected from the IUT to conform to the base specification.

5.3.2 TP naming conventions

Each TP is given a unique identification. This unique identification is built up to contain the following string of information:

TP_<group>_<iut>_<x>_<nn>

TP to indicate that it is a test purpose;

<group> which group TP belongs to;

<iut> type of IUT (i.e. FE or BE);

X type of testing [i.e. Behaviour Valid tests (BV) or Behaviour Invalid tests (BI)];

<nn> sequential TP number (01–99).

The naming conventions are as described in [Table 3](#).

Table 3 — TP naming convention

Identifier: TP/<group>/<iut>/<x>-<nn>		
<group>		
<i>applicable for FE</i>	CR	Charge Report
<i>applicable for FE</i>	BEF	Back End Feedback
<i>applicable for BE</i>	CRR	Charge Report Response
<iut> = type of IUT		
	FE	Front End
	BE	Back End
x = Type of testing		
	BV	Behaviour Valid tests
	BI	Behaviour Invalid tests
<nn> = sequential number		
	(01–99)	test purpose number

5.4 Conformance test report

The supplier of the Front End and Back End, respectively, is responsible for providing a conformance test report.

The supplier of the Front End shall complete the proforma conformance test report (PCTR) for Front End as defined in [Annex D](#).

The supplier of the Back End shall complete the proforma conformance test report (PCTR) for Back End as defined in [Annex E](#).

Annex A (normative)

Test purposes (TP) for Front End

A.1 General

This annex contains the test purposes (TP) for the conformity evaluation of Front End to ISO 17575-1.

A.2 TP symbols conventions

A special notation and symbol convention shall be used as defined in what follows.

Symbols are used in the description of the TPs with meanings according to [Table A.1](#).

Table A.1 — Description of TP symbols

Symbol	Description
$XXX.rq \Rightarrow$	The Tester sends the $XXX.rq$ to the IUT
$\Leftarrow YYY.rs$	The IUT sends the $YYY.rs$ to the Tester
$\Leftarrow YYY.rs = \{\text{attribute1, attribute2, attribute3}\}$	The IUT sends the $YYY.rs$ to the Tester. $YYY.rs$ shall not consist of any attributes different than attribute1, attribute2, attribute3. If any of attributes in the list is optional, it may be missing in $YYY.rs$.
$\Leftarrow YYY.rs = \{\text{attribute1} = \text{value1}\}$	The IUT sends the $YYY.rs$ to the Tester with attribute1. Value of attribute1, i.e. value1 shall be stored by the tester and will be utilized in further TP steps.
$A \equiv B$	A “is equal to” B
$A \rightarrow B$	A “is transformed” into B
$\emptyset$	Means “empty” or “not set”
$A \mid B$	A OR B
$x \rightarrow n^-$	Value of parameter x is very close to n and x is less than n
$x \rightarrow n^+$	Value of parameter x is very close to n and x greater than n

In addition, the sequence of ADUs issued by the Front End is not constrained by ISO 17575-1. This means that ADU cannot in general be forced to be generated by the IUT. To execute the test purposes it may be needed to filter out some ADUs, as they might not be applicable for TP, e.g. some ADUs are applicable for different toll context. This is illustrated in [Figure A.1](#).

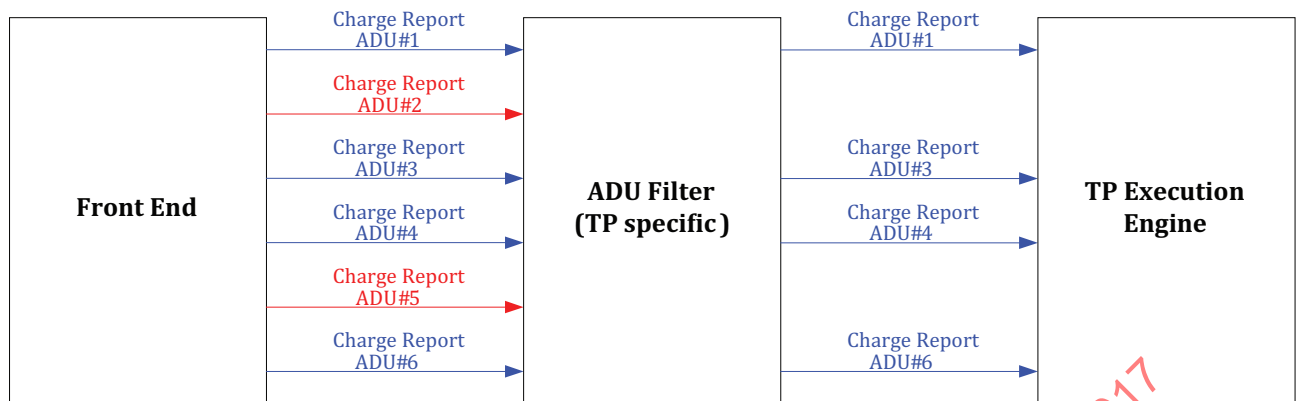


Figure A.1 — Handling of ADUs applicable for particular TP

A.3 Charge report

A.3.1 General

These test purposes apply to data elements in Charge Report as claimed in ISO 17575-1:2016:

- Table B.6/1 – ChargeReport;
- Table B.6/3 – UsageStatement;
- Table B.7/1 – AuthenticatedChargeReport;
- Table B.7/2 – protocolVersion;
- Table B.7/3 – obeId;
- Table B.7/4 – vehicleLPNr;
- Table B.7/8 – chargeReportFinalRecipient;
- Table B.7/9 – timeOfReport;
- Table B.7/10 – reportPeriod;
- Table B.7/15 – chargeReportCounter;
- Table B.7/16 – mileage;
- Table B.9/1 – usageStatementID.

NOTE No test purposes for Behaviour Invalid are specified (BI) as ISO 17575-1 does not specify any Behaviour Invalid of Front End.

A.3.2 BV test purposes

The test subgroup objective is to test the behaviour of the IUT with respect to data elements contained in Charge Report.

Table A.2 — TP_CR_FE_BV_01

TP_CR_FE_BV_01		Protocol version handling	
TP origin		Specific	
Reference		ISO 17575-1:2016, 7.8.1	
Initial condition		Front End is initialized and has a toll context activated.	
		Front End sends context data in the format defined by ISO 17575-1:2016.	
		Authentication is not used.	
Stimulus and expected behaviour			
	IUT		Tester
1	ChargeReport = { protocolVersion, obeId, vehicleLPNr, paymentMeans, serviceProviderContract, tollContext chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, usageStatementList, sumVatForThisSession, accountStatus, chargeReportCounter, mileage, listOfCCCAttributes }	⇒	
2			Verify the structure of sent ChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1 .
3			IF verify NOT “OK” THEN TP failed
4			Verify that protocolVersion is ISO 7575-1:2016
5			IF verify “OK” THEN TP passed ELSE TP failed ENDIF
6		⇐	ChargeReportResponse = { chargeReportRespSender = any, dataReceived = ChargeReport.timeofReport ChargeReport.mileage ChargeReport.chargeReportCounter, versionsResponse = ∅, obeStatusForDriver = 0, accountUpdate = ∅ }

Table A.3 — TP_CR_FE_BV_02

TP_CR_FE_BV_02		Verify that Time of Report is according to the local time	
TP origin		Specific	
Reference		ISO 17575-1:2016, 7.3.1	
Initial condition		Front End received already Context Data with correctly set timeZone attribute and requesting presence of timeOfReport. Front End is initialized and has a toll context activated. Authentication is not used.	
Stimulus and expected behaviour			
	IUT		Tester
1	ChargeReport = { protocolVersion, obeId, vehicleLPNr, paymentMeans, serviceProviderContract, tollContext, chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, usageStatementList, sumVatForThisSession, accountStatus, chargeReportCounter, mileage, listOfCCCAttributes }	⇒	
2			Verify the structure of sent ChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1 .
3			IF verify NOT “OK” THEN TP failed
4			Verify that timeofReport is according to the local time
5			IF verify “OK” THEN TP passed ELSE TP failed ENDIF
6		⇐	ChargeReportResponse = { chargeReportRespSender = any, dataReceived = ChargeReport.timeofReport ChargeReport.mileage ChargeReport.chargeReportCounter, versionsResponse = ∅, obeStatusForDriver = 0, accountUpdate = ∅ }

Table A.4 — TP_CR_FE_BV_03

TP_CR_FE_BV_03	Verify that Charge Report Counter is incremented		
TP origin	Specific		
Reference	ISO 17575-1:2016, 7.3.4		
Initial condition	Front End received already Context Data requesting presence of chargeReportCounter. Front End is initialized and has a toll context activated. Authentication is not used.		
Stimulus and expected behaviour			
	IUT		Tester
1	ChargeReport = { protocolVersion, obeId, vehicleLPNr, paymentMeans, serviceProviderContract, tollContext, chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, usageStatementList, sumVatForThisSession, accountStatus, chargeReportCounter = v1, mileage, listOfCCCAttributes }	⇒	
2			Verify the structure of sent ChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1 .
3			IF verify NOT "OK" THEN TP failed
4		⇐	ChargeReportResponse = { chargeReportRespSender = any, dataReceived = v1, versionsResponse = ∅, obeStatusForDriver = 0, accountUpdate = ∅ }
5	ChargeReport = { protocolVersion, obeId, vehicleLPNr, paymentMeans, serviceProviderContract, tollContext, chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, usageStatementList, sumVatForThisSession, accountStatus, chargeReportCounter = v2, mileage, listOfCCCAttributes }	⇒	

Table A.4 (continued)

6			Verify the structure of sent ChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1 .
7			IF verify NOT "OK" THEN TP failed
8			IF v2 > v1 THEN TP passed ELSE TP failed ENDIF
9		⇐	ChargeReportResponse = { chargeReportRespSender = any, dataReceived = v2, versionsResponse = ∅, obeStatusForDriver = 0, accountUpdate = ∅ }

Table A.5 — TP_CR_FE_BV_04

TP_CR_FE_BV_04	Charge Report Counter Overflow		
TP origin	Specific		
Reference	ISO 17575-1:2016, 7.3.4		
Initial condition	Front End received already Context Data requesting presence of chargeReportCounter. Front End is initialized and has a toll context activated. Next charge Report Counter value is a maximum value (i.e. 4294967295) Authentication is not used.		
Stimulus and expected behaviour			
	IUT		Tester
1	ChargeReport = { protocolVersion, obId, vehicleLPNr, paymentMeans, serviceProviderContract, tollContext, chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, usageStatementList, sumVatForThisSession, accountStatus, chargeReportCounter = 4294967295, mileage, listOfCCCAttributes }	⇒	
2			Verify the structure of sent ChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1 .
3			IF verify NOT “OK” THEN TP failed

Table A.5 (continued)

4		⇐	ChargeReportResponse = { chargeReportRespSender = any, dataReceived = 4294967295, versionsResponse = ∅, obeStatusForDriver = 0, accountUpdate = ∅ }
5	ChargeReport = { protocolVersion, obeld, vehicleLPNr, paymentMeans, serviceProviderContract, tollContext, chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, usageStatementList, sumVatForThisSession, accountStatus, chargeReportCounter = v1, mileage, listOfCCCAttributes }	⇒	
6			Verify the structure of sent ChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1 .
7			IF verify NOT "OK" THEN TP failed
8			IF v1 equals to 0 THEN TP passed ELSE TP failed ENDIF
9		⇐	ChargeReportResponse = { chargeReportRespSender = any, dataReceived = v1, versionsResponse = ∅, obeStatusForDriver = 0, accountUpdate = ∅ }

Table A.6 — TP_CR_FE_BV_05

TP_CR_FE_BV_05	Verify that mileage is not decreasing		
TP origin	Specific		
Reference	ISO 17575-1:2016, 7.3.5		
Initial condition	Front End received already Context Data requesting presence of mileage. Front End is initialized and has a toll context activated. Authentication is not used.		
Stimulus and expected behaviour			
	IUT		Tester
1	ChargeReport = { protocolVersion, obeId, vehicleLPNr, paymentMeans, serviceProviderContract, tollContext, chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, usageStatementList, sumVatForThisSession, accountStatus, chargeReportCounter, mileage = v1, listOfCCCAttributes }	⇒	
2			Verify the structure of sent ChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1 .
3			IF verify NOT “OK” THEN TP failed
4		⇐	ChargeReportResponse = { chargeReportRespSender = any, dataReceived = v1, versionsResponse = ∅, obeStatusForDriver = 0, accountUpdate = ∅ }

Table A.6 (continued)

5	ChargeReport = { protocolVersion, obeId, vehicleLPNr, paymentMeans, serviceProviderContract, tollContext, chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, usageStatementList, sumVatForThisSession, accountStatus, chargeReportCounter = v2, mileage, listOfCCCAttributes }	⇒	
6			Verify the structure of sent ChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1 .
7			IF verify NOT "OK" THEN TP failed
8			IF v2 – v1 ≥ 0 THEN TP passed ELSE TP failed ENDIF
9		⇐	ChargeReportResponse = { chargeReportRespSender = any, dataReceived = v2, versionsResponse = ∅, obeStatusForDriver = 0, accountUpdate = ∅ }

Table A.7 — TP_CR_FE_BV_06

TP_CR_FE_BV_06	Verify that mileage counter restarts from zero in case of overflow		
TP origin	Specific		
Reference	ISO 17575-1:2016, 7.3.5		
Initial condition	Front End received already Context Data requesting presence of mileage. Front End is initialized and has a toll context activated. Mileage.dist is about to roll-over, e.g. its value is close to 4294967295. Authentication is not used.		
Stimulus and expected behaviour			
	IUT		Tester
1	ChargeReport = { protocolVersion, obeId, vehicleLPNr, paymentMeans, serviceProviderContract, tollContext, chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, usageStatementList, sumVatForThisSession, accountStatus, chargeReportCounter, mileage → 4294967295-, listOfCCCAttributes }	⇒	
2			Verify the structure of sent ChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1 .
3			IF verify NOT “OK” THEN TP failed
4		⇐	ChargeReportResponse = { chargeReportRespSender = any, dataReceived = ChargeReport.mileage, versionsResponse = ∅, obeStatusForDriver = 0, accountUpdate = ∅ }
5	Mileage has changed enough to overflow		

Table A.7 (continued)

6	ChargeReport = { protocolVersion, obeId, vehicleLPNr, paymentMeans, serviceProviderContract, tollContext, chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, usageStatementList, sumVatForThisSession, accountStatus, chargeReportCounter = v1, mileage, listOfCCCAttributes }	⇒	
7			Verify the structure of sent ChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1 .
8			IF verify NOT "OK" THEN TP failed
9			IF v1 → 0+ (i.e. mileage counts again from zero being close to zero value) THEN TP passed ELSE TP failed ENDIF
10		⇐	ChargeReportResponse = { chargeReportRespSender = any, dataReceived = v1, versionsResponse = ∅, obeStatusForDriver = 0, accountUpdate = ∅ }

Table A.8 — TP_CR_FE_BV_07

TP_CR_FE_BV_07	Verify that OBE ID is fixed		
TP origin	Specific		
Reference	ISO 17575-1:2016, 7.5.1		
Initial condition	Front End received already Context Data requesting presence of obeld. Front End is initialized and has a toll context activated. Authentication is not used.		
Stimulus and expected behaviour			
	IUT		Tester
1	ChargeReport = { protocolVersion, obeId, vehicleLPNr, paymentMeans, serviceProviderContract, tollContext, chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, usageStatementList, sumVatForThisSession, accountStatus, chargeReportCounter, mileage, listOfCCCAttributes }	⇒	
2			Verify the structure of sent ChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1 .
3			IF verify NOT “OK” THEN TP failed
4		⇐	ChargeReportResponse = { chargeReportRespSender = any, dataReceived = ChargeReport.timeofReport ChargeReport.mileage ChargeReport.chargeReportCounter, versionsResponse = ∅, obeStatusForDriver = 0, accountUpdate = ∅ }

Table A.8 (continued)

5	ChargeReport = { protocolVersion, obId, vehicleLPNr, paymentMeans, serviceProviderContract, tollContext, chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, usageStatementList, sumVatForThisSession, accountStatus, chargeReportCounter, mileage, listOfCCCAttributes }	⇒	
6			Verify the structure of sent ChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1 .
7			IF verify NOT "OK" THEN TP failed
8		⇐	ChargeReportResponse = { chargeReportRespSender = any, dataReceived = ChargeReport.timeofReport ChargeReport.mileage ChargeReport.chargeReportCounter, versionsResponse = ∅, obeStatusForDriver = 0, accountUpdate = ∅ }
...			
n	ChargeReport = { protocolVersion, obId, vehicleLPNr, paymentMeans, serviceProviderContract, tollContext, chargeReportFinalRecipient, timeOfReport, vreportPeriod, versionInfo, usageStatementList, sumVatForThisSession, accountStatus, chargeReportCounter, mileage, listOfCCCAttributes }	⇒	
			Verify the structure of sent ChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1 .

Table A.8 (continued)

			IF verify NOT "OK" THEN TP failed
			IF (obeld the same in every ChargeReport) THEN TP passed ELSE TP failed ENDIF
		⇐	ChargeReportResponse = { chargeReportRespSender = any, dataReceived = ChargeReport.timeofReport ChargeReport.mileage ChargeReport.chargeReportCounter, versionsResponse = ∅, obeStatusForDriver = 0, accountUpdate = ∅ }

Table A.9 — TP_CR_FE_BV_08

TP_CR_FE_BV_08	Verify that Vehicle License Plate Number is not used in Charge Report once it is used in lower level data elements		
TP origin	Specific		
Reference	ISO 17575-1:2016, 7.5.2		
Initial condition	Front End received already Context Data requesting presence of vehicleLPNr. Front End is initialized and has a toll context activated. Authentication is not used.		
Stimulus and expected behaviour			
	IUT		Tester
1	ChargeReport = { protocolVersion, obId, vehicleLPNr, paymentMeans, serviceProviderContract, tollContext, chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, usageStatementList, sumVatForThisSession, accountStatus, chargeReportCounter, mileage, listOfCCCAttributes }	⇒	
2			Verify the structure of sent ChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1 .

Table A.9 (continued)

3			IF verify NOT "OK" THEN TP failed
4			IF (vehicleLPNr present in usageStatementList at any level) AND (vehicleLPNr present in ChargeReport) THEN TP failed ELSE TP passed ENDIF
5		⇐	ChargeReportResponse = { chargeReportRespSender = any, dataReceived = ChargeReport.timeofReport ChargeReport.mileage ChargeReport.chargeReportCounter, versionsResponse = ∅, obeStatusForDriver = 0, accountUpdate = ∅ }

Table A.10 — TP_CR_FE_BV_09

TP_CR_FE_BV_09	Verify the uniqueness of Usage Statement ID		
TP origin	Specific		
Reference	ISO 17575-1:2016, 7.6.3		
Initial condition	Front End received already Context Data. Front End is initialized and has a toll context activated. Authentication is not used.		
Stimulus and expected behaviour			
	IUT		Tester
1	ChargeReport = { protocolVersion, obId, vehicleLPNr, paymentMeans, serviceProviderContract, tollContext, chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, usageStatementList, sumVatForThisSession, accountStatus, chargeReportCounter, mileage, listOfCCCAttributes }	⇒	
2			Verify the structure of sent ChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1 .
3			IF verify NOT “OK” THEN TP failed

Table A.10 (continued)

4			IF each usageStatementId unique within usageStatementList THEN TP passed ELSE TP failed ENDIF
5			IF verify "OK" THEN TP passed ELSE TP failed ENDIF
6		⇐	ChargeReportResponse = { chargeReportRespSender = any, dataReceived = ChargeReport.timeofReport ChargeReport.mileage ChargeReport.chargeReportCounter, versionsResponse = ∅, obeStatusForDriver = 0, accountUpdate = ∅ }

Table A.11 — TP_CR_FE_BV_10

TP_CR_FE_BV_10	Verify that begin period is lower than end period in Report Period		
TP origin	Specific		
Reference	ISO 17575-1:2016, 7.3.2		
Initial condition	Front End received already Context Data requesting presence of reportPeriod. Front End is initialized and has a toll context activated. Authentication is not used.		
Stimulus and expected behaviour			
	IUT		Tester
1	ChargeReport = { protocolVersion, obId, vehicleLPN, paymentMeans, serviceProviderContract, tollContext, chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, usageStatementList, sumVatForThisSession, accountStatus, chargeReportCounter, mileage, listOfCCCAttributes }	⇒	
2			Verify the structure of sent ChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1 .

Table A.11 (continued)

3			IF verify NOT "OK" THEN TP failed
4			IF (reportPeriod.beginOfPeriod < reportPeriod.endOfPeriod) THEN TP passed ELSE TP failed ENDIF
5		⇐	ChargeReportResponse = { chargeReportRespSender = any, dataReceived = ChargeReport.timeofReport ChargeReport.mileage ChargeReport.chargeReportCounter, versionsResponse = ∅, obeStatusForDriver = 0, accountUpdate = ∅ }

Table A.12 — TP_CR_FE_BV_16

TP_CR_FE_BV_16	Verify that chargeReportFinalRecipient is the same in Context Data and Charge Report		
TP origin	Specific		
Reference	ISO 17575-1:2016, 7.5.6		
Initial condition	Front End received already Context Data: — requesting presence of chargeReportFinalRecipient in tollContextOverview, usageStatement and chargeObjectId, and — consisting of chargeReportFinalRecipient (which equals to v1) in tollContextPartitionOverview. Front End is initialized and has a toll context activated. Authentication is not used.		
Stimulus and expected behaviour			
	IUT		Tester
1	ChargeReport = { protocolVersion, obeId, vehicleLPNr, paymentMeans, serviceProviderContract, tollContext, chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, usageStatementList, sumVatForThisSession, accountStatus, chargeReportCounter, mileage, listOfCCCAttributes }	⇒	
2			Verify the structure of sent ChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1 .
3			IF verify NOT “OK” THEN TP failed

Table A.12 (continued)

4			Verify that all chargeReportFinalRecipient elements used in ChargeReport equals to v1
5			IF verify "OK" THEN TP passed ELSE TP failed ENDIF
6		⇐	ChargeReportResponse = { chargeReportRespSender = any, dataReceived = ChargeReport.timeofReport ChargeReport.mileage ChargeReport.chargeReportCounter, versionsResponse = ∅, obeStatusForDriver = 0, accountUpdate = ∅ }

Table A.13 — TP_CR_FE_BV_17

TP_CR_FE_BV_17	Verify that Authenticated Charge Report is used		
TP origin	Specific		
Reference	ISO 17575-1:2016, 7.2.1 and 7.4.1		
Initial condition	Front End received already Context Data. Front End is initialized and has a toll context activated. Authentication is used using Authenticated Charge Report.		
Stimulus and expected behaviour			
	IUT		Tester
1	AuthenticatedChargeReport = { chargeReportPer, messageAuthenticator }	⇒	
2			Verify the structure of sent AuthenticatedChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.2 .
3			IF verify NOT “OK” THEN TP failed ELSE TP passed ENDIF

Table A.14 — TP_CR_FE_BV_18

TP_CR_FE_BV_18	Verify that authenticated usage statement is used		
TP origin	Specific		
Reference	ISO 17575-1:2016, 7.4.3		
Initial condition	Front End received already Context Data. Front End is initialized and has a toll context activated. Authentication is used for Usage Statement within a Charge Report.		
Stimulus and expected behaviour			
	IUT		Tester
1	ChargeReport = { protocolVersion, obeld, vehicleLPNr, paymentMeans, serviceProviderContract, tollContext, chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, authenticatedUsageStatement, sumVatForThisSession, accountStatus, chargeReportCounter, mileage, listOfCCCAttributes }	⇒	
2			Verify the structure of sent ChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.3 .
3			IF verify NOT “OK” THEN TP failed ELSE TP passed ENDIF

A.3.3 BI test purposes

No BI test purposes are applicable for this TP group.

A.4 Back End feedback

A.4.1 General

These test purposes apply to Back End feedback sent in Charge Report Response as claimed in ISO 17575-1:2016:

- Table B.6/2 – ChargeReportResponse;
- Table B.8/5 – obeStatusForDriver;
- Table B.8/6 – accountUpdate;

— Table B.10/1 to 5 – account types: credit, distance, time, duration and events.

NOTE No test purposes for Behaviour Invalid are specified (BI) as ISO 17575-1 does not specify any Behaviour Invalid of Front End.

A.4.2 BV test purposes

The test subgroup objective is

- to test IUT behaviour with respect to OBE Status for Driver,
- to test the IUT behaviour with respect to account update handling for the following types of on-board accounts:
 - credit,
 - distance,
 - time,
 - duration, and
 - event.

Table A.15 — TP_BEF_FE_BV_01

TP_BEF_FE_BV_01	Verify that chargeReportFinalRecipient is the same in Context Data and Charge Report		
TP origin	Specific		
Reference	ISO 17575-1:2016, 7.5.7		
Initial condition	Front End received already Context Data requesting presence of mileage or chargeReportCounter or timeofReport. Front End is initialized and has a toll context activated. Authentication is not used.		
Stimulus and expected behaviour			
	IUT		Tester
1	ChargeReport = { protocolVersion, obeId, vehicleLPNr, paymentMeans, serviceProviderContract tollContext, chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, usageStatementList, sumVatForThisSession, vaccountStatus, chargeReportCounter, mileage, listOfCCCAttributes }	⇒	
2			Verify the structure of sent ChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1 .
3			IF verify NOT “OK” THEN TP failed

Table A.15 (continued)

4		⇐	ChargeReportResponse = { chargeReportRespSender = any, dataReceived = ChargeReport.timeofReport ChargeReport.mileage ChargeReport.chargeReportCounter, versionsResponse = ∅, obeStatusForDriver = 0, accountUpdate = ∅ }
5			Verify that OBE status is "OK"
6			IF verify "OK" THEN TP passed ELSE TP failed ENDIF

Table A.16 — TP_BEF_FE_BV_02

TP_BEF_FE_BV_02	Verify that OBE Status For Driver is “NOK” having received Charge Report Response		
TP origin	Specific		
Reference	ISO 17575-1:2016, 7.5.7		
Initial condition	Front End received already Context Data requesting presence of mileage or chargeReportCounter or timeofReport. Front End is initialized and has a toll context activated. Authentication is not used.		
Stimulus and expected behaviour			
	IUT		Tester
1	ChargeReport = { protocolVersion, obeld, vehicleLPNr, paymentMeans, serviceProviderContract, tollContext, chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, usageStatementList, sumVatForThisSession, accountStatus, chargeReportCounter, mileage, listOfCCCAttributes }	⇒	
2			Verify the structure of sent ChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1 .
3			IF verify NOT “OK” THEN TP failed

Table A.16 (continued)

4		⇐	ChargeReportResponse = { chargeReportRespSender = any, dataReceived = ChargeReport.timeofReport ChargeReport.mileage ChargeReport.chargeReportCounter, versionsResponse = ∅, obeStatusForDriver = 1, accountUpdate = ∅ }
5			Verify that OBE status is 'NOK'
6			IF verify "OK" THEN TP passed ELSE TP failed ENDIF

Table A.17 — TP_BEF_FE_BV_03

TP_BEF_FE_BV_03	Verify that OBE status for Driver is “contactOperator” having received Charge Report Response		
TP origin	Specific		
Reference	ISO 17575-1:2016, 7.5.7		
Initial condition	Front End received already Context Data requesting presence of mileage or chargeReportCounter or timeofReport. Front End is initialized and has a toll context activated. Authentication is not used.		
Stimulus and expected behaviour			
	IUT		Tester
1	ChargeReport = { protocolVersion, obeId, vehicleLPNr, paymentMeans, serviceProviderContract, tollContext, chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, usageStatementList, sumVatForThisSession, accountStatus, chargeReportCounter, mileage, listOfCCCAttributes }	⇒	
2			Verify the structure of sent ChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1 .
3			IF verify NOT “OK” THEN TP failed

Table A.17 (continued)

4		⇐	ChargeReportResponse = { chargeReportRespSender = any, dataReceived = ChargeReport.timeofReport ChargeReport.mileage ChargeReport.chargeReportCounter, versionsResponse = ∅, obeStatusForDriver = 2, accountUpdate = ∅ }
5			Verify that OBE status is "contactOperator"
6			IF verify "OK" THEN TP passed ELSE TP failed ENDIF

Table A.18 — TP_BEFE_BV_05

TP_BEFE_BV_05	Verify that OBE status for Driver is “noSignalling” having received Charge Report Response		
TP origin	Specific		
Reference	ISO 17575-1:2016, 7.5.7		
Initial condition	Front End received already Context Data requesting presence of mileage or chargeReportCounter or timeofReport. Front End is initialized and has a toll context activated. Authentication is not used.		
Stimulus and expected behaviour			
	IUT		Tester
1	ChargeReport = { protocolVersion, obeld, vehicleLPNr, paymentMeans, serviceProviderContract, tollContext, chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, usageStatementList, sumVatForThisSession, accountStatus, chargeReportCounter, mileage, listOfCCCAttributes }	⇒	
2			Verify the structure of sent ChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1 .
3			IF verify NOT “OK” THEN TP failed

Table A.18 (continued)

4		⇐	ChargeReportResponse = { chargeReportRespSender = any, dataReceived = ChargeReport.timeofReport ChargeReport.mileage ChargeReport.chargeReportCounter, versionsResponse = ∅, obeStatusForDriver = 255, accountUpdate = ∅ }
5			Verify that OBE status is “contactOperator”
6			IF verify “noSignalling” THEN TP passed ELSE TP failed ENDIF

Table A.19 — TP_BEFE_BV_06

TP_BEFE_BV_06	Verify that DUT sets Account Status to “OK” for credit based on-board accounts		
TP origin	Specific		
Reference	ISO 17575-1:2016, 7.7.1		
Initial condition	Front End received already Context Data: — applicable for on-board accounts, — requesting presence of mileage or chargeReportCounter or timeofReport, and — requesting presence of accountStatus. Front End implementation supports on-board account. On-board account is credit based with unit defined as UNIT. Threshold for on-board account is set to THR. Front End is initialized and has a toll context activated. Authentication is not used.		
Stimulus and expected behaviour			
	IUT		Tester
1	ChargeReport = { protocolVersion, obeld, vehicleLPNr, paymentMeans, serviceProviderContract, tollContext, chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, usageStatementList, sumVatForThisSession, accountStatus = accountS1, chargeReportCounter, mileage, listOfCCCAttributes }	⇒	

Table A.19 (continued)

2			Verify the structure of sent ChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1 .
3			IF verify NOT "OK" THEN TP failed
4		←	ChargeReportResponse = { chargeReportRespSender = any, dataReceived = ChargeReport.timeofReport ChargeReport.mileage ChargeReport.chargeReportCounter, versionsResponse = ∅, obeStatusForDriver = ∅, accountUpdate = accountU } where accountU = setAccount { newCreditLimit = { paymentFeeAmount > THR, paymentFeeUnit = UNIT }, newDistanceLimit = ∅, newTimeLimit = ∅, newDurationLimit = ∅, newEventLimit = ∅, } }
5	ChargeReport = { protocolVersion, obeld, vehicleLPNr, paymentMeans, serviceProviderContract, tollContext, chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, usageStatementList, sumVatForThisSession, accountStatus = accountS2, chargeReportCounter, mileage, listOfCCCAttributes } No credit change shall occur between step 4 and step 5.	⇒	
6			Verify the structure of sent ChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1 .

Table A.19 (continued)

7			IF verify NOT "OK" THEN TP failed
8			IF (accountS2 equals to 0 (OK)) THEN TP passed ELSE TP failed ENDIF
9		⇐	ChargeReportResponse = { chargeReportRespSender = any, dataReceived = ChargeReport.timeofReport ChargeReport.mileage ChargeReport.chargeReportCounter, versionsResponse = ∅, obeStatusForDriver = ∅, accountUpdate = ∅ }

Table A.20 — TP_BEF_FE_BV_07

TP_BEF_FE_BV_07	Verify that DUT sets Account Status to “LOW” for credit based on-board accounts		
TP origin	Specific		
Reference	ISO 17575-1:2016, 7.7.1		
Initial condition	Front End received already Context Data: — applicable for on-board accounts, — requesting presence of mileage or chargeReportCounter or timeofReport, and — requesting presence of accountStatus. Front End implementation supports on-board account. On-board account is credit based with unit defined as UNIT. Threshold for on-board account is set to THR. Front End is initialized and has a toll context activated. Authentication is not used.		
Stimulus and expected behaviour			
	IUT		Tester
1	ChargeReport = { protocolVersion, obeld, vehicleLPNr, paymentMeans, serviceProviderContract, tollContext, chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, usageStatementList, sumVatForThisSession, accountStatus = accountS1, chargeReportCounter, mileage, listOfCCCAttributes }	⇒	

Table A.20 (continued)

2			Verify the structure of sent ChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1 .
3			IF verify NOT "OK" THEN TP failed
4		←	ChargeReportResponse = { chargeReportRespSender = any, dataReceived = ChargeReport.timeofReport ChargeReport.mileage ChargeReport.chargeReportCounter, versionsResponse = ∅, obeStatusForDriver = ∅, accountUpdate = accountU } where accountU = setAccount { newCreditLimit = { 0 < paymentFeeAmount < THR, paymentFeeUnit = UNIT }, newDistanceLimit = ∅, newTimeLimit = ∅, newDurationLimit = ∅, newEventLimit = ∅, } }
5	ChargeReport = { protocolVersion, obeld, vehicleLPNr, paymentMeans, serviceProviderContract, tollContext, chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, usageStatementList, sumVatForThisSession, accountStatus = accountS2, chargeReportCounter, mileage, listOfCCCAttributes } No credit change shall occur between step 4 and step 5.	⇒	
6			Verify the structure of sent ChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1 .

Table A.20 (continued)

7			IF verify NOT "OK" THEN TP failed
8			IF (accountS2 equals to 1 (low)) THEN TP passed ELSE TP failed ENDIF
9		⇐	ChargeReportResponse = { chargeReportRespSender = any, dataReceived = ChargeReport.timeofReport ChargeReport.mileage ChargeReport.chargeReportCounter, versionsResponse = ∅, obeStatusForDriver = ∅, accountUpdate = ∅ }

Table A.21 — TP_BEF_FE_BV_08

TP_BEF_FE_BV_08	Verify that DUT sets Account Status to “EMPTY” for credit based on-board accounts		
TP origin	Specific		
Reference	ISO 17575-1:2016, 7.7.1		
Initial condition	Front End received already Context Data: — applicable for on-board accounts, — requesting presence of mileage or chargeReportCounter or timeofReport, and — requesting presence of accountStatus. Front End implementation supports on-board account. On-board account is credit based with unit defined as UNIT. Threshold for on-board account is set to THR. Front End is initialized and has a toll context activated. Authentication is not used.		
Stimulus and expected behaviour			
	IUT		Tester
1	ChargeReport = { protocolVersion, obeld, vehicleLPNr, paymentMeans, serviceProviderContract, tollContext, chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, usageStatementList, sumVatForThisSession, accountStatus = accountS1, chargeReportCounter, mileage, listOfCCCAttributes }	⇒	

Table A.21 (continued)

2			Verify the structure of sent ChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1 .
3			IF verify NOT "OK" THEN TP failed
4		←	ChargeReportResponse = { chargeReportRespSender = any, dataReceived = ChargeReport.timeofReport ChargeReport.mileage ChargeReport.chargeReportCounter, versionsResponse = ∅, obeStatusForDriver = ∅, accountUpdate = accountU } where accountU = setAccount { newCreditLimit = { paymentFeeAmount = 0, paymentFeeUnit = UNIT }, newDistanceLimit = ∅, newTimeLimit = ∅, newDurationLimit = ∅, newEventLimit = ∅, } }
5	ChargeReport = { protocolVersion, obeld, vehicleLPNr, paymentMeans, serviceProviderContract, tollContext, chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, usageStatementList, sumVatForThisSession, accountStatus = accountS2, chargeReportCounter, mileage, listOfCCCAttributes } No credit change shall occur between step 4 and step 5.	⇒	
6			Verify the structure of sent ChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1 .

Table A.21 (continued)

7			IF verify NOT "OK" THEN TP failed
8			IF (accountS2 equals to 2 (empty)) THEN TP passed ELSE TP failed ENDIF
9		⇐	ChargeReportResponse = { chargeReportRespSender = any, dataReceived = ChargeReport.timeofReport ChargeReport.mileage ChargeReport.chargeReportCounter, versionsResponse = ∅, obeStatusForDriver = ∅, accountUpdate = ∅ }

Table A.22 — TP_BEFEV_09

TP_BEFEV_09	Verify that DUT sets Account Status to “NEGATIVE” for credit based on-board accounts		
TP origin	Specific		
Reference	ISO 17575-1:2016, 7.7.1		
Initial condition	Front End received already Context Data: — applicable for on-board accounts, — requesting presence of mileage or chargeReportCounter or timeofReport, and — requesting presence of accountStatus. Front End implementation supports on-board account. On-board account is credit based with unit defined as UNIT. Threshold for on-board account is set to THR. Front End is initialized and has a toll context activated. Authentication is not used.		
Stimulus and expected behaviour			
	IUT		Tester
1	ChargeReport = { protocolVersion, obeld, vehicleLPNr, paymentMeans, serviceProviderContract, tollContext, chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, usageStatementList, sumVatForThisSession, accountStatus = accountS1, chargeReportCounter, mileage, listOfCCCAttributes }	⇒	

Table A.22 (continued)

2			Verify the structure of sent ChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1 .
3			IF verify NOT "OK" THEN TP failed
4		←	ChargeReportResponse = { chargeReportRespSender = any, dataReceived = ChargeReport.timeofReport ChargeReport.mileage ChargeReport.chargeReportCounter, versionsResponse = ∅, obeStatusForDriver = ∅, accountUpdate = accountU } where accountU = setAccount { newCreditLimit = { paymentFeeAmount < 0, paymentFeeUnit = UNIT }, newDistanceLimit = ∅, newTimeLimit = ∅, newDurationLimit = ∅, newEventLimit = ∅, } }
5	ChargeReport = { protocolVersion, obeld, vehicleLPNr, paymentMeans, serviceProviderContract, tollContext, chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, usageStatementList, sumVatForThisSession, accountStatus = accountS2, chargeReportCounter, mileage, listOfCCCAttributes } No credit change shall occur between step 4 and step 5.	⇒	
6			Verify the structure of sent ChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1 .

Table A.22 (continued)

7			IF verify NOT "OK" THEN TP failed
8			IF (accountS2 equals to 3 (negative)) THEN TP passed ELSE TP failed ENDIF
9		⇐	ChargeReportResponse = { chargeReportRespSender = any, dataReceived = ChargeReport.timeofReport ChargeReport.mileage ChargeReport.chargeReportCounter, versionsResponse = ∅, obeStatusForDriver = ∅, accountUpdate = ∅ }

Table A.23 — TP_BEF_FE_BV_10

TP_BEF_FE_BV_10	Verify that DUT sets Account Status to “OK” for distance based on-board accounts		
TP origin	Specific		
Reference	ISO 175751:2016, 7.7.1		
Initial condition	Front End received already Context Data: — applicable for on-board accounts, — requesting presence of mileage or chargeReportCounter or timeofReport, and — requesting presence of accountStatus. Front End implementation supports on-board account. On-board account is distance based with unit defined as UNIT. Threshold for on-board account is set to THR. Front End is initialized and has a toll context activated. Authentication is not used.		
Stimulus and expected behaviour			
	IUT		Tester
1	ChargeReport = { protocolVersion, obeld, vehicleLPNr, paymentMeans, serviceProviderContract, tollContext, chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, usageStatementList, sumVatForThisSession, accountStatus = accountS1, chargeReportCounter, mileage, listOfCCCAttributes }	⇒	

Table A.23 (continued)

2			Verify the structure of sent ChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1 .
3			IF verify NOT "OK" THEN TP failed
4		←	ChargeReportResponse = { chargeReportRespSender = any, dataReceived = ChargeReport.timeofReport ChargeReport.mileage ChargeReport.chargeReportCounter, versionsResponse = ∅, obeStatusForDriver = ∅, accountUpdate = accountU } where accountU = setAccount { newCreditLimit = ∅, newDistanceLimit = { dist > THR, disUnit = UNIT }, newTimeLimit = ∅, newDurationLimit = ∅, newEventLimit = ∅, } }
5	ChargeReport = { protocolVersion, obeld, vehicleLPNr, paymentMeans, serviceProviderContract, tollContext, chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, usageStatementList, sumVatForThisSession, accountStatus = accountS2, chargeReportCounter, mileage, listOfCCCAttributes } No distance change shall occur between step 4 and step 5.	⇒	
6			Verify the structure of sent ChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1 .

Table A.23 (continued)

7			IF verify NOT "OK" THEN TP failed
8			IF (accountS2 equals to 0 (OK)) THEN TP passed ELSE TP failed ENDIF
9		⇐	ChargeReportResponse = { chargeReportRespSender = any, dataReceived = ChargeReport.timeofReport ChargeReport.mileage ChargeReport.chargeReportCounter, versionsResponse = ∅, obeStatusForDriver = ∅, accountUpdate = ∅ }

Table A.24 — TP_BEF_FE_BV_11

TP_BEF_FE_BV_11	Verify that DUT sets Account Status to “LOW” for distance based on-board accounts		
TP origin	Specific		
Reference	ISO 17575-1:2016, 7.7.1		
Initial condition	Front End received already Context Data: — applicable for on-board accounts, — requesting presence of mileage or chargeReportCounter or timeofReport, and — requesting presence of accountStatus. Front End implementation supports on-board account. On-board account is distance based with unit defined as UNIT. Threshold for on-board account is set to THR. Front End is initialized and has a toll context activated. Authentication is not used.		
Stimulus and expected behaviour			
	IUT		Tester
1	ChargeReport = { protocolVersion, obeld, vehicleLPNr, paymentMeans, serviceProviderContract, tollContext, chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, usageStatementList, sumVatForThisSession, accountStatus = accountS1, chargeReportCounter, mileage, listOfCCCAttributes }	⇒	

Table A.24 (continued)

2			Verify the structure of sent ChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1 .
3			IF verify NOT "OK" THEN TP failed
4		←	ChargeReportResponse = { chargeReportRespSender = any, dataReceived = ChargeReport.timeofReport ChargeReport.mileage ChargeReport.chargeReportCounter, versionsResponse = ∅, obeStatusForDriver = ∅, accountUpdate = accountU } where accountU = setAccount { newCreditLimit = ∅, newDistanceLimit = { 0 < dist < THR, disUnit = UNIT }, newTimeLimit = ∅, newDurationLimit = ∅, newEventLimit = ∅, } }
5	ChargeReport = { protocolVersion, obeld, vehicleLPNr, paymentMeans, serviceProviderContract, tollContext, chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, usageStatementList, sumVatForThisSession, accountStatus = accountS2, chargeReportCounter, mileage, listOfCCCAttributes } No distance change shall occur between step 4 and step 5.	⇒	
6			Verify the structure of sent ChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1 .

Table A.24 (continued)

7			IF verify NOT "OK" THEN TP failed
8			IF (accountS2 equals to 1 (low)) THEN TP passed ELSE TP failed ENDIF
9		⇐	ChargeReportResponse = { chargeReportRespSender = any, dataReceived = ChargeReport.timeofReport ChargeReport.mileage ChargeReport.chargeReportCounter, versionsResponse = ∅, obeStatusForDriver = ∅, accountUpdate = ∅ }

Table A.25 — TP_BEFEV_12

TP_BEFEV_12	Verify that DUT sets Account Status to “EMPTY” for distance based on-board accounts		
TP origin	Specific		
Reference	ISO 17575-1:2016, 7.7.1		
Initial condition	Front End received already Context Data: — applicable for on-board accounts, — requesting presence of mileage or chargeReportCounter or timeofReport, and — requesting presence of accountStatus. Front End implementation supports on-board account. On-board account is distance based with unit defined as UNIT. Threshold for on-board account is set to THR. Front End is initialized and has a toll context activated. Authentication is not used.		
Stimulus and expected behaviour			
	IUT		Tester
1	ChargeReport = { protocolVersion, obeld, vehicleLPNr, paymentMeans, serviceProviderContract, tollContext, chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, usageStatementList, sumVatForThisSession, accountStatus = accountS1, chargeReportCounter, mileage, listOfCCCAttributes }	⇒	

Table A.25 (continued)

2			Verify the structure of sent ChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1 .
3			IF verify NOT "OK" THEN TP failed
4		←	ChargeReportResponse = { chargeReportRespSender = any, dataReceived = ChargeReport.timeofReport ChargeReport.mileage ChargeReport.chargeReportCounter, versionsResponse = ∅, obeStatusForDriver = ∅, accountUpdate = accountU } where accountU = setAccount { newCreditLimit = ∅, newDistanceLimit = { dist = 0, disUnit = UNIT }, newTimeLimit = ∅, newDurationLimit = ∅, newEventLimit = ∅, } }
5	ChargeReport = { protocolVersion, obeld, vehicleLPNr, paymentMeans, serviceProviderContract, tollContext, chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, usageStatementList, sumVatForThisSession, accountStatus = accountS2, chargeReportCounter, mileage, listOfCCCAttributes } No distance change shall occur between step 4 and step 5.	⇒	
6			Verify the structure of sent ChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1 .

Table A.25 (continued)

7			IF verify NOT "OK" THEN TP failed
8			IF (accountS2 equals to 2 (empty)) THEN TP passed ELSE TP failed ENDIF
9		⇐	ChargeReportResponse = { chargeReportRespSender = any, dataReceived = ChargeReport.timeofReport ChargeReport.mileage ChargeReport.chargeReportCounter, versionsResponse = ∅, obeStatusForDriver = ∅, accountUpdate = ∅ }

Table A.26 — TP_BEFE_BV_13

TP_BEFE_BV_13	Verify that DUT sets Account Status to “NEGATIVE” for distance based on-board accounts		
TP origin	Specific		
Reference	ISO 17575-1:2016, 7.7.1		
Initial condition	Front End received already Context Data: — applicable for on-board accounts, — requesting presence of mileage or chargeReportCounter or timeofReport, and — requesting presence of accountStatus. Front End implementation supports on-board account. On-board account is distance based with unit defined as UNIT. Threshold for on-board account is set to THR. Front End is initialized and has a toll context activated. Authentication is not used.		
Stimulus and expected behaviour			
	IUT		Tester
1	ChargeReport = { protocolVersion, obeld, vehicleLPNr, paymentMeans, serviceProviderContract, tollContext, chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, usageStatementList, sumVatForThisSession, accountStatus = accountS1, chargeReportCounter, mileage, listOfCCCAttributes }	⇒	

Table A.26 (continued)

2			Verify the structure of sent ChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1 .
3			IF verify NOT "OK" THEN TP failed
4		←	ChargeReportResponse = { chargeReportRespSender = any, dataReceived = ChargeReport.timeofReport ChargeReport.mileage ChargeReport.chargeReportCounter, versionsResponse = ∅, obeStatusForDriver = ∅, accountUpdate = accountU } where accountU = setAccount { newCreditLimit = ∅, newDistanceLimit = { dist < 0, disUnit = UNIT }, newTimeLimit = ∅, newDurationLimit = ∅, newEventLimit = ∅, } }
5	ChargeReport = { protocolVersion, obeld, vehicleLPNr, paymentMeans, serviceProviderContract, tollContext, chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, usageStatementList, sumVatForThisSession, accountStatus = accountS2, chargeReportCounter, mileage, listOfCCCAttributes } No distance change shall occur between step 4 and step 5.	⇒	
6			Verify the structure of sent ChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1 .

Table A.26 (continued)

7			IF verify NOT "OK" THEN TP failed
8			IF (accountS2 equals to 3 (negative)) THEN TP passed ELSE TP failed ENDIF
9		⇐	ChargeReportResponse = { chargeReportRespSender = any, dataReceived = ChargeReport.timeofReport ChargeReport.mileage ChargeReport.chargeReportCounter, versionsResponse = ∅, obeStatusForDriver = ∅, accountUpdate = ∅ }

Table A.27 — TP_BEF_FE_BV_14

TP_BEF_FE_BV_14	Verify that DUT sets Account Status to “OK” for time based on-board accounts		
TP origin	Specific		
Reference	ISO 17575-1:2016, 7.7.1		
Initial condition	Front End received already Context Data: — applicable for on-board accounts, — requesting presence of mileage or chargeReportCounter or timeofReport, and — requesting presence of accountStatus. Front End implementation supports on-board account. On-board account is time based. Threshold for on-board account is set to THR. Front End is initialized and has a toll context activated. Authentication is not used.		
Stimulus and expected behaviour			
	IUT		Tester
1	ChargeReport = { protocolVersion, obeld, vehicleLPNr, paymentMeans, serviceProviderContract, tollContext, chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, usageStatementList, sumVatForThisSession, accountStatus = accountS1, chargeReportCounter, mileage, listOfCCCAttributes }	⇒	

Table A.27 (continued)

2			Verify the structure of sent ChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1 .
3			IF verify NOT "OK" THEN TP failed
4		⇐	ChargeReportResponse = { chargeReportRespSender = any, dataReceived = ChargeReport.timeofReport ChargeReport.mileage ChargeReport.chargeReportCounter, versionsResponse = ∅, obeStatusForDriver = ∅, accountUpdate = accountU } where accountU = setAccount { newCreditLimit = ∅, newDistanceLimit = ∅, newTimeLimit = currentTime + THR, newDurationLimit = ∅, newEventLimit = ∅, }
5	ChargeReport = { protocolVersion, obeld, vehicleLPNr, paymentMeans, serviceProviderContract, tollContext, chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, usageStatementList, sumVatForThisSession, accountStatus = accountS2, chargeReportCounter, mileage, listOfCCCAttributes } This data unit shall be sent not later than DUT changes its accountStatus.	⇒	
6			Verify the structure of sent ChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1 .

Table A.27 (continued)

7			IF verify NOT "OK" THEN TP failed
8			IF (accountS2 equals to 0 (OK)) THEN TP passed ELSE TP failed ENDIF
9		⇐	ChargeReportResponse = { chargeReportRespSender = any, dataReceived = ChargeReport.timeofReport ChargeReport.mileage ChargeReport.chargeReportCounter, versionsResponse = ∅, obeStatusForDriver = ∅, accountUpdate = ∅ }

Table A.28 — TP_BEFE_BV_15

TP_BEFE_BV_15	Verify that DUT sets Account Status to “LOW” for time based on-board accounts		
TP origin	Specific		
Reference	ISO 17575-1:2016, 7.7.1		
Initial condition	Front End received already Context Data: — applicable for on-board accounts, — requesting presence of mileage or chargeReportCounter or timeofReport, and — requesting presence of accountStatus. Front End implementation supports on-board account. On-board account is time based. Threshold for on-board account is set to THR. Front End is initialized and has a toll context activated. Authentication is not used.		
Stimulus and expected behaviour			
	IUT		Tester
1	ChargeReport = { protocolVersion, obeld, vehicleLPNr, paymentMeans, serviceProviderContract, tollContext, chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, usageStatementList, sumVatForThisSession, accountStatus = accountS1, chargeReportCounter, mileage, listOfCCCAttributes }	⇒	

Table A.28 (continued)

2			Verify the structure of sent ChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1 .
3			IF verify NOT "OK" THEN TP failed
4		⇐	ChargeReportResponse= { chargeReportRespSender = any, dataReceived = ChargeReport.timeofReport ChargeReport.mileage ChargeReport.chargeReportCounter, versionsResponse = ∅, obeStatusForDriver = ∅, accountUpdate = accountU } where accountU = setAccount { newCreditLimit = ∅, newDistanceLimit = ∅, currentTime < < newTimeLimit < currentTime + THR, newDurationLimit = ∅, newEventLimit = ∅, }
5	ChargeReport = { protocolVersion, obeId, vehicleLPNr, paymentMeans, serviceProviderContract, tollContext, chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, usageStatementList, sumVatForThisSession, accountStatus = accountS2, chargeReportCounter, mileage, listOfCCCAttributes } This data unit shall be sent not later than DUT changes its accountStatus.	⇒	
6			Verify the structure of sent ChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1 .

Table A.28 (continued)

7			IF verify NOT "OK" THEN TP failed
8			IF (accountS2 equals to 1 (low)) THEN TP passed ELSE TP failed ENDIF
9		⇐	ChargeReportResponse = { chargeReportRespSender = any, dataReceived = ChargeReport.timeofReport ChargeReport.mileage ChargeReport.chargeReportCounter, versionsResponse = ∅, obeStatusForDriver = ∅, accountUpdate = ∅ }

Table A.29 — TP_BEF_FE_BV_16

TP_BEF_FE_BV_16	Verify that DUT sets Account Status to “EMPTY” for time based on-board accounts		
TP origin	Specific		
Reference	ISO 17575-1:2016, 7.7.1		
Initial condition	Front End received already Context Data: — applicable for on-board accounts, — requesting presence of mileage or chargeReportCounter or timeofReport, and — requesting presence of accountStatus. Front End implementation supports on-board account. On-board account is time based. Threshold for on-board account is set to THR. Front End is initialized and has a toll context activated. Authentication is not used.		
Stimulus and expected behaviour			
	IUT		Tester
1	ChargeReport = { protocolVersion, obeld, vehicleLPNr, paymentMeans, serviceProviderContract, tollContext, chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, usageStatementList, sumVatForThisSession, accountStatus = accountS1, chargeReportCounter, mileage, listOfCCCAttributes }	⇒	

Table A.29 (continued)

2			Verify the structure of sent ChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1 .
3			IF verify NOT "OK" THEN TP failed
4		⇐	ChargeReportResponse = { chargeReportRespSender = any, dataReceived = ChargeReport.timeofReport ChargeReport.mileage ChargeReport.chargeReportCounter, versionsResponse = ∅, obeStatusForDriver = ∅, accountUpdate = accountU } where accountU = setAccount { newCreditLimit = ∅, newDistanceLimit = ∅, newTimeLimit = currentTime, newDurationLimit = ∅, newEventLimit = ∅, }
5	ChargeReport = { protocolVersion, obeld, vehicleLPNr, paymentMeans, serviceProviderContract, tollContext, chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, usageStatementList, sumVatForThisSession, accountStatus = accountS2, chargeReportCounter, mileage, listOfCCCAttributes } This data unit shall be sent not later than DUT changes its accountStatus.	⇒	
6			Verify the structure of sent ChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1 .

Table A.29 (continued)

7			IF verify NOT "OK" THEN TP failed
8			IF (accountS2 equals to 2 (empty)) THEN TP passed ELSE TP failed ENDIF
9		⇐	ChargeReportResponse = { chargeReportRespSender = any, dataReceived = ChargeReport.timeofReport ChargeReport.mileage ChargeReport.chargeReportCounter, versionsResponse = ∅, obeStatusForDriver = ∅, accountUpdate = ∅ }

Table A.30 — TP_BEF_FE_BV_17

TP_BEF_FE_BV_17	Verify that DUT sets Account Status to "NEGATIVE" for time based on-board accounts		
TP origin	Specific		
Reference	ISO 17575-1:2016, 7.7.1		
Initial condition	Front End received already Context Data: — applicable for on-board accounts, — requesting presence of mileage or chargeReportCounter or timeofReport, and — requesting presence of accountStatus. Front End implementation supports on-board account. On-board account is time based. Threshold for on-board account is set to THR. Front End is initialized and has a toll context activated. Authentication is not used.		
Stimulus and expected behaviour			
	IUT		Tester
1	ChargeReport = { protocolVersion, obeld, vehicleLPNr, paymentMeans, serviceProviderContract, tollContext, chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, usageStatementList, sumVatForThisSession, accountStatus = accountS1, chargeReportCounter, mileage, listOfCCCAttributes }	⇒	

Table A.30 (continued)

2			Verify the structure of sent ChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1 .
3			IF verify NOT "OK" THEN TP failed
4		⇐	ChargeReportResponse = { chargeReportRespSender = any, dataReceived = ChargeReport.timeofReport ChargeReport.mileage ChargeReport.chargeReportCounter, versionsResponse = ∅, obeStatusForDriver = ∅, accountUpdate = accountU } where accountU = setAccount { newCreditLimit = ∅, newDistanceLimit = ∅, newTimeLimit < currentTime, newDurationLimit = ∅, newEventLimit = ∅, }
5	ChargeReport = { protocolVersion, obeld, vehicleLPNr, paymentMeans, serviceProviderContract, tollContext, chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, usageStatementList, sumVatForThisSession, accountStatus = accountS2, chargeReportCounter, mileage, listOfCCCAttributes } This data unit shall be sent not later than DUT changes its accountStatus.	⇒	
6			Verify the structure of sent ChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1 .

Table A.30 (continued)

7			IF verify NOT "OK" THEN TP failed
8			IF (accountS2 equals to 3 (negative)) THEN TP passed ELSE TP failed ENDIF
9		⇐	ChargeReportResponse = { chargeReportRespSender = any, dataReceived = ChargeReport.timeofReport ChargeReport.mileage ChargeReport.chargeReportCounter, versionsResponse = ∅, obeStatusForDriver = ∅, accountUpdate = ∅ }

Table A.31 — TP_BEF_FE_BV_18

TP_BEF_FE_BV_18	Verify that DUT sets Account Status to “OK” for duration based on-board accounts		
TP origin	Specific		
Reference	ISO 17575-1:2016, 7.7.1		
Initial condition	Front End received already Context Data: — applicable for on-board accounts, — requesting presence of mileage or chargeReportCounter or timeofReport, and — requesting presence of accountStatus. Front End implementation supports on-board account. On-board account is duration based with unit defined as UNIT. Threshold for on-board account is set to THR. Front End is initialized and has a toll context activated. Authentication is not used.		
Stimulus and expected behaviour			
	IUT		Tester
1	ChargeReport = { protocolVersion, obeld, vehicleLPNr, paymentMeans, serviceProviderContract, tollContext, chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, usageStatementList, sumVatForThisSession, accountStatus = accountS1, chargeReportCounter, mileage, listOfCCCAttributes }	⇒	

Table A.31 (continued)

2			Verify the structure of sent ChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1 .
3			IF verify NOT "OK" THEN TP failed
4		←	<pre> ChargeReportResponse = { chargeReportRespSender = any, dataReceived = ChargeReport.timeofReport ChargeReport.mileage ChargeReport.chargeReportCounter, versionsResponse = ∅, obeStatusForDriver = ∅, accountUpdate = accountU } where accountU = setAccount { newCreditLimit = ∅, newDistanceLimit = ∅, newTimeLimit = ∅, newDurationLimit = { dur > THR, durUnit = UNIT }, newEventLimit = ∅, } </pre>
5	ChargeReport = { protocolVersion, obeld, vehicleLPNr, paymentMeans, serviceProviderContract, tollContext, chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, usageStatementList, sumVatForThisSession, accountStatus = accountS2, chargeReportCounter, mileage, listOfCCCAttributes }	⇒	
6			Verify the structure of sent ChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1 .

Table A.31 (continued)

7			IF verify NOT "OK" THEN TP failed
8			IF (accountS2 equals to 0 (OK)) THEN TP passed ELSE TP failed ENDIF
9		⇐	ChargeReportResponse = { chargeReportRespSender = any, dataReceived = ChargeReport.timeofReport ChargeReport.mileage ChargeReport.chargeReportCounter, versionsResponse = ∅, obeStatusForDriver = ∅, accountUpdate = ∅ }

Table A.32 — TP_BEF_FE_BV_19

TP_BEF_FE_BV_19	Verify that DUT sets Account Status to “LOW” for duration based on-board accounts		
TP origin	Specific		
Reference	ISO 17575-1:2016, 7.7.1		
Initial condition	Front End received already Context Data: — applicable for on-board accounts, — requesting presence of mileage or chargeReportCounter or timeofReport, and — requesting presence of accountStatus. Front End implementation supports on-board account. On-board account is duration based with unit defined as UNIT. Threshold for on-board account is set to THR. Front End is initialized and has a toll context activated. Authentication is not used.		
Stimulus and expected behaviour			
	IUT		Tester
1	ChargeReport = { protocolVersion, obeld, vehicleLPNr, paymentMeans, serviceProviderContract, tollContext, chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, usageStatementList, sumVatForThisSession, accountStatus = accountS1, chargeReportCounter, mileage, listOfCCCAttributes }	⇒	

Table A.32 (continued)

2			Verify the structure of sent ChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1 .
3			IF verify NOT "OK" THEN TP failed
4		←	ChargeReportResponse = { chargeReportRespSender = any, dataReceived = ChargeReport.timeofReport ChargeReport.mileage ChargeReport.chargeReportCounter, versionsResponse = ∅, obeStatusForDriver = ∅, accountUpdate = accountU } where accountU = setAccount { newCreditLimit = ∅, newDistanceLimit = ∅, newTimeLimit = ∅, newDurationLimit = { 0 < dur < THR, durUnit = UNIT }, newEventLimit = ∅, } }
5	ChargeReport = { protocolVersion, obeld, vehicleLPNr, paymentMeans, serviceProviderContract, tollContext, chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, usageStatementList, sumVatForThisSession, accountStatus = accountS2, chargeReportCounter, mileage, listOfCCCAttributes } This data unit shall be sent not later than DUT changes its accountStatus.	⇒	
6			Verify the structure of sent ChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1 .

Table A.32 (continued)

7			IF verify NOT "OK" THEN TP failed
8			IF (accountS2 equals to 1 (low)) THEN TP passed ELSE TP failed ENDIF
9		⇐	ChargeReportResponse = { chargeReportRespSender = any, dataReceived = ChargeReport.timeofReport ChargeReport.mileage ChargeReport.chargeReportCounter, versionsResponse = ∅, obeStatusForDriver = ∅, accountUpdate = ∅ }

Table A.33 — TP_BEFE_BV_20

TP_BEFE_BV_20	Verify that DUT sets Account Status to “EMPTY” for duration based on-board accounts		
TP origin	Specific		
Reference	ISO 17575-1:2016, 7.7.1		
Initial condition	Front End received already Context Data: — applicable for on-board accounts, — requesting presence of mileage or chargeReportCounter or timeofReport, and — requesting presence of accountStatus. Front End implementation supports on-board account. On-board account is duration based with unit defined as UNIT. Threshold for on-board account is set to THR. Front End is initialized and has a toll context activated. Authentication is not used.		
Stimulus and expected behaviour			
	IUT		Tester
1	ChargeReport = { protocolVersion, obeld, vehicleLPNr, paymentMeans, serviceProviderContract, tollContext, chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, usageStatementList, sumVatForThisSession, accountStatus = accountS1, chargeReportCounter, mileage, listOfCCCAttributes }	⇒	

Table A.33 (continued)

2			Verify the structure of sent ChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1 .
3			IF verify NOT "OK" THEN TP failed
4		←	ChargeReportResponse = { chargeReportRespSender = any, dataReceived = ChargeReport.timeofReport ChargeReport.mileage ChargeReport.chargeReportCounter, versionsResponse = ∅, obeStatusForDriver = ∅, accountUpdate = accountU } where accountU = setAccount { newCreditLimit = ∅, newDistanceLimit = ∅, newTimeLimit = ∅, newDurationLimit = { dur = 0, durUnit = UNIT }, newEventLimit = ∅, }
5	ChargeReport = { protocolVersion, obeld, vehicleLPNr, paymentMeans, serviceProviderContract, tollContext, chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, usageStatementList, sumVatForThisSession, accountStatus = accountS2, chargeReportCounter, mileage, listOfCCCAttributes } This data unit shall be sent not later than DUT changes its accountStatus.	⇒	
6			Verify the structure of sent ChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1 .

Table A.33 (continued)

7			IF verify NOT "OK" THEN TP failed
8			IF (accountS2 equals to 2 (empty)) THEN TP passed ELSE TP failed ENDIF
9		⇐	ChargeReportResponse= { chargeReportRespSender = any, dataReceived = ChargeReport.timeofReport ChargeReport.mileage ChargeReport.chargeReportCounter, versionsResponse = ∅, obeStatusForDriver = ∅, accountUpdate = ∅ }

Table A.34 — TP_BEFE_BV_21

TP_BEFE_BV_21	Verify that DUT sets Account Status to "NEGATIVE" for duration based on-board accounts		
TP origin	Specific		
Reference	ISO 17575-1:2016, 7.7.1		
Initial condition	Front End received already Context Data: — applicable for on-board accounts, — requesting presence of mileage or chargeReportCounter or timeofReport, and — requesting presence of accountStatus. Front End implementation supports on-board account. On-board account is is duration based with unit defined as UNIT. Threshold for on-board account is set to THR. Front End is initialized and has a toll context activated. Authentication is not used.		
Stimulus and expected behaviour			
	IUT		Tester
1	ChargeReport = { protocolVersion, obeld, vehicleLPNr, paymentMeans, serviceProviderContract, tollContext, chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, usageStatementList, sumVatForThisSession, accountStatus = accountS1, chargeReportCounter, mileage, listOfCCCAttributes }	⇒	

Table A.34 (continued)

2			Verify the structure of sent ChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1 .
3			IF verify NOT "OK" THEN TP failed
4		←	<pre> ChargeReportResponse = { chargeReportRespSender = any, dataReceived = ChargeReport.timeofReport ChargeReport.mileage ChargeReport.chargeReportCounter, versionsResponse = ∅, obeStatusForDriver = ∅, accountUpdate = accountU } where accountU = setAccount { newCreditLimit = ∅, newDistanceLimit = ∅, newTimeLimit = ∅, newDurationLimit = { dur ≤ 0, durUnit = UNIT }, newEventLimit = ∅, } </pre>
5	ChargeReport = { protocolVersion, obeld, vehicleLPNr, paymentMeans, serviceProviderContract, tollContext, chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, usageStatementList, sumVatForThisSession, accountStatus = accountS2, chargeReportCounter, mileage, listOfCCCAttributes }	⇒	
6			Verify the structure of sent ChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1 .

Table A.34 (continued)

7			IF verify NOT "OK" THEN TP failed
8			IF (accountS2 equals to 3 (negative)) THEN TP passed ELSE TP failed ENDIF
9		⇐	ChargeReportResponse = { chargeReportRespSender = any, dataReceived = ChargeReport.timeofReport ChargeReport.mileage ChargeReport.chargeReportCounter, versionsResponse = ∅, obeStatusForDriver = ∅, accountUpdate = ∅ }

Table A.35 — TP_BEFE_BV_22

TP_BEFE_BV_22	Verify that DUT sets Account Status to “OK” for event based on-board accounts		
TP origin	Specific		
Reference	ISO 17575-1:2016, 7.7.1		
Initial condition	Front End received already Context Data: — applicable for on-board accounts, — requesting presence of mileage or chargeReportCounter or timeofReport, and — requesting presence of accountStatus. Front End implementation supports on-board account. On-board account is event based. Threshold for on-board account is set to THR. Front End is initialized and has a toll context activated. Authentication is not used.		
Stimulus and expected behaviour			
	IUT		Tester
1	ChargeReport = { protocolVersion, obeld, vehicleLPNr, paymentMeans, serviceProviderContract, tollContext, chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, usageStatementList, sumVatForThisSession, accountStatus = accountS1, chargeReportCounter, mileage, listOfCCCAttributes }	⇒	

Table A.35 (continued)

2			Verify the structure of sent ChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1 .
3			IF verify NOT "OK" THEN TP failed
4		⇐	ChargeReportResponse = { chargeReportRespSender = any, dataReceived = ChargeReport.timeofReport ChargeReport.mileage ChargeReport.chargeReportCounter, versionsResponse = ∅, obeStatusForDriver = ∅, accountUpdate = accountU } where accountU = setAccount { newCreditLimit = ∅, newDistanceLimit = ∅, newTimeLimit = ∅, newDurationLimit = ∅, newEventLimit > THR, }
5	ChargeReport = { protocolVersion, obeld, vehicleLPNr, paymentMeans, serviceProviderContract, tollContext, chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, usageStatementList, sumVatForThisSession, accountStatus = accountS2, chargeReportCounter, mileage, listOfCCCAttributes } No accountable events shall occur between step 4 and step 5.	⇒	
6			Verify the structure of sent ChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1 .

Table A.35 (continued)

7			IF verify NOT "OK" THEN TP failed
8			IF (accountS2 equals to 0 (OK)) THEN TP passed ELSE TP failed ENDIF
9		⇐	ChargeReportResponse = { chargeReportRespSender = any, dataReceived = ChargeReport.timeofReport ChargeReport.mileage ChargeReport.chargeReportCounter, versionsResponse = ∅, obeStatusForDriver = ∅, accountUpdate = ∅ }

Table A.36 — TP_BEF_FE_BV_23

TP_BEF_FE_BV_23	Verify that DUT sets Account Status to “LOW” for event based on-board accounts		
TP origin	Specific		
Reference	ISO 17575-1:2016, 7.7.1		
Initial condition	Front End received already Context Data: — applicable for on-board accounts, — requesting presence of mileage or chargeReportCounter or timeofReport, and — requesting presence of accountStatus. Front End implementation supports on-board account. On-board account is event based. Threshold for on-board account is set to THR. Front End is initialized and has a toll context activated. Authentication is not used.		
Stimulus and expected behaviour			
	IUT		Tester
1	ChargeReport = { protocolVersion, obeld, vehicleLPNr, paymentMeans, serviceProviderContract, tollContext, chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, usageStatementList, sumVatForThisSession, accountStatus = accountS1, chargeReportCounter, mileage, listOfCCCAttributes }	⇒	

Table A.36 (continued)

2			Verify the structure of sent ChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1 .
3			IF verify NOT "OK" THEN TP failed
4		⇐	ChargeReportResponse = { chargeReportRespSender = any, dataReceived = ChargeReport.timeofReport ChargeReport.mileage ChargeReport.chargeReportCounter, versionsResponse = ∅, obeStatusForDriver = ∅, accountUpdate = accountU } where accountU = setAccount { newCreditLimit = ∅ newDistanceLimit = ∅, newTimeLimit = ∅, newDurationLimit = ∅, 0 < newEventLimit < THR, }
5	ChargeReport = { protocolVersion, obeld, vehicleLPNr, paymentMeans, serviceProviderContract, tollContext, chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, usageStatementList, sumVatForThisSession, accountStatus = accountS2, chargeReportCounter, mileage, listOfCCCAttributes } No accountable events shall occur between step 4 and step 5.	⇒	
6			Verify the structure of sent ChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1 .

Table A.36 (continued)

7			IF verify NOT "OK" THEN TP failed
8			IF (accountS2 equals to 1 (low)) THEN TP passed ELSE TP failed ENDIF
9		⇐	ChargeReportResponse = { chargeReportRespSender = any, dataReceived = ChargeReport.timeofReport ChargeReport.mileage ChargeReport.chargeReportCounter, versionsResponse = ∅, obeStatusForDriver = ∅, accountUpdate = ∅ }

Table A.37 — TP_BEF_FE_BV_24

TP_BEF_FE_BV_24	Verify that DUT sets Account Status to “EMPTY” for event based on-board accounts		
TP origin	Specific		
Reference	ISO 17575-1:2016, 7.7.1		
Initial condition	Front End received already Context Data: — applicable for on-board accounts, — requesting presence of mileage or chargeReportCounter or timeofReport, and — requesting presence of accountStatus. Front End implementation supports on-board account. On-board account is event based. Threshold for on-board account is set to THR. Front End is initialized and has a toll context activated. Authentication is not used.		
Stimulus and expected behaviour			
	IUT		Tester
1	ChargeReport = { protocolVersion, obeld, vehicleLPNr, paymentMeans, serviceProviderContract, tollContext, chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, usageStatementList, sumVatForThisSession, accountStatus = accountS1, chargeReportCounter, mileage, listOfCCCAttributes }	⇒	

Table A.37 (continued)

2			Verify the structure of sent ChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1 .
3			IF verify NOT "OK" THEN TP failed
4		←	ChargeReportResponse = { chargeReportRespSender = any, dataReceived = ChargeReport.timeofReport ChargeReport.mileage ChargeReport.chargeReportCounter, versionsResponse = ∅, obeStatusForDriver = ∅, accountUpdate = accountU } where accountU = setAccount { newCreditLimit = ∅ newDistanceLimit = ∅, newTimeLimit = ∅, newDurationLimit = ∅, newEventLimit = 0, }
5	ChargeReport = { protocolVersion, obeld, vehicleLPNr, paymentMeans, serviceProviderContract, tollContext, chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, usageStatementList, sumVatForThisSession, accountStatus = accountS2, chargeReportCounter, mileage, listOfCCCAttributes } No accountable events shall occur between step 4 and step 5.	⇒	
6			Verify the structure of sent ChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1 .

Table A.37 (continued)

7			IF verify NOT "OK" THEN TP failed
8			IF (accountS2 equals to 2 (empty)) THEN TP passed ELSE TP failed ENDIF
9		⇐	ChargeReportResponse = { chargeReportRespSender = any, dataReceived = ChargeReport.timeofReport ChargeReport.mileage ChargeReport.chargeReportCounter, versionsResponse = ∅, obeStatusForDriver = ∅, accountUpdate = ∅ }

Table A.38 — TP_BEFE_BV_25

TP_BEFE_BV_25	Verify that DUT sets Account Status to “NEGATIVE” for event based on-board accounts		
TP origin	Specific		
Reference	ISO 17575-1:2016, 7.7.1		
Initial condition	Front End received already Context Data: — applicable for on-board accounts, — requesting presence of mileage or chargeReportCounter or timeofReport, and — requesting presence of accountStatus. Front End implementation supports on-board account. On-board account is event based. Threshold for on-board account is set to THR. Front End is initialized and has a toll context activated. Authentication is not used.		
Stimulus and expected behaviour			
	IUT		Tester
1	ChargeReport = { protocolVersion, obeld, vehicleLPNr, paymentMeans, serviceProviderContract, tollContext, chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, usageStatementList, sumVatForThisSession, accountStatus = accountS1, chargeReportCounter, mileage, listOfCCCAttributes }	⇒	

Table A.38 (continued)

2			Verify the structure of sent ChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1 .
3			IF verify NOT "OK" THEN TP failed
4		←	ChargeReportResponse = { chargeReportRespSender = any, dataReceived = ChargeReport.timeofReport ChargeReport.mileage ChargeReport.chargeReportCounter, versionsResponse = ∅, obeStatusForDriver = ∅, accountUpdate = accountU } where accountU = setAccount { newCreditLimit = ∅ newDistanceLimit = ∅, newTimeLimit = ∅, newDurationLimit = ∅, newEventLimit < 0, }
5	ChargeReport = { protocolVersion, obeld, vehicleLPNr, paymentMeans, serviceProviderContract, tollContext, chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, usageStatementList, sumVatForThisSession, accountStatus = accountS2, chargeReportCounter, mileage, listOfCCCAttributes } No accountable events shall occur between step 4 and step 5.	⇒	
6			Verify the structure of sent ChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1 .

Table A.38 (continued)

7			IF verify NOT "OK" THEN TP failed
8			IF (accountS2 equals to 3 (negative)) THEN TP passed ELSE TP failed ENDIF
9		⇐	ChargeReportResponse = { chargeReportRespSender = any, dataReceived = ChargeReport.timeofReport ChargeReport.mileage ChargeReport.chargeReportCounter, versionsResponse = ∅, obeStatusForDriver = ∅, accountUpdate = ∅ }

Table A.39 — TP_BEF_FE_BV_26

TP_BEF_FE_BV_26	Verify that DUT updates Account Status for credit based on-board accounts upon AddToAccount operation
TP origin	Specific
Reference	ISO 17575-1:2016, 7.7.5
Initial condition	Front End received already Context Data: — applicable for on-board accounts, — requesting presence of mileage or chargeReportCounter or timeofReport, and — requesting presence of accountStatus. Front End implementation supports on-board account. On-board account is credit based with unit defined as UNIT. Threshold for on-board account is set to THR. The Front End shall have the value of accountStatus set other than "ok". Front End is initialized and has a toll context activated. Front End shall not utilize any credits during the whole test purpose. Authentication is not used.

Table A.39 (continued)

Stimulus and expected behaviour			
	IUT		Tester
1	ChargeReport = { protocolVersion, obeId, vehicleLPNr, paymentMeans, serviceProviderContract, tollContext, chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, usageStatementList, sumVatForThisSession, accountStatus = accountS1, chargeReportCounter, mileage, listOfCCCAttributes }	⇒	
2			Verify the structure of sent ChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1 .
3			IF verify NOT "OK" THEN TP failed
4		⇐	ChargeReportResponse = { chargeReportRespSender = any, dataReceived = ChargeReport.timeofReport ChargeReport.mileage ChargeReport.chargeReportCounter, versionsResponse = ∅, obeStatusForDriver = ∅, accountUpdate = accountU } where accountU = addToAccount { addCredit = { 0 < paymentFeeAmount < THR, paymentFeeUnit = UNIT }, addDistance = ∅, addTime = ∅, addDuration = ∅, addEvents = ∅, }

Table A.39 (continued)

5	ChargeReport = { protocolVersion, obeld, vehicleLPNr, paymentMeans, serviceProviderContract, tollContext, chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, usageStatementList, sumVatForThisSession, accountStatus = accountS2, chargeReportCounter, mileage, listOfCCCAttributes }	⇒	
6			Verify the structure of sent ChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1 .
7			IF verify NOT "OK" THEN TP failed
8			The following transitions from accountS1 to accountS2 are allowed (in notation accountS1 → account S2): 3 (negative) → 3 (negative) 3 (negative) → 2 (empty) 3 (negative) → 1 (low) 2 (empty) → 1 (low) 1 (low) → 1 (low) 1 (low) → 0 (ok) 0 (ok) → 0 (ok) IF (transition not allowed) THEN TP failed ENDIF
9			IF (accountS2 equals to 0 (ok)) THEN TP passed ELE { accountS1 = accountS2 GOTO STEP 4 ENDIF
10		⇐	ChargeReportResponse = { chargeReportRespSender = any, dataReceived = ChargeReport.timeofReport ChargeReport.mileage ChargeReport.chargeReportCounter, versionsResponse = ∅, obeStatusForDriver = ∅, accountUpdate = ∅ } TP passed

Table A.40 — TP_BEF_FE_BV_27

TP_BEF_FE_BV_27	Verify that DUT updates Account Status for distance based on-board accounts upon AddToAccount operation		
TP origin	Specific		
Reference	ISO 17575-1:2016, 7.7.5		
Initial condition	Front End received already Context Data: — applicable for on-board accounts, — requesting presence of mileage or chargeReportCounter or timeofReport, and — requesting presence of accountStatus. Front End implementation supports on-board account. On-board account is distance based with unit defined as UNIT. Threshold for on-board account is set to THR. The Front End shall have the value of accountStatus set other than “ok”. Front End is initialized and has a toll context activated. Front End shall not change its mileage during the whole test purpose. Authentication is not used.		
Stimulus and expected behaviour			
	IUT		Tester
1	ChargeReport = { protocolVersion, obeId, vehicleLPNr, paymentMeans, serviceProviderContract, tollContext, chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, usageStatementList, sumVatForThisSession, accountStatus = accountS1, chargeReportCounter, mileage, listOfCCCAttributes }	⇒	
2			Verify the structure of sent ChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1 .
3			IF verify NOT “OK” THEN TP failed

Table A.40 (continued)

4		⇐	ChargeReportResponse = { chargeReportRespSender = any, dataReceived = ChargeReport.timeofReport ChargeReport.mileage ChargeReport.chargeReportCounter, versionsResponse = ∅, obeStatusForDriver = ∅, accountUpdate = accountU } where accountU = addToAccount { addCredit = ∅, addDistance = { 0 < dist < THR, disUnit = UNIT }, addTime = ∅, addDuration = ∅, addEvents = ∅, }
5	ChargeReport = { protocolVersion, obeId, vehicleLPNr, paymentMeans, serviceProviderContract, tollContext, chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, usageStatementList, sumVatForThisSession, accountStatus = accountS2, chargeReportCounter, mileage, listOfCCCAttributes }	⇒	
6			Verify the structure of sent ChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1 .
7			IF verify NOT "OK" THEN TP failed

Table A.40 (continued)

8			The following transitions from accountS1 to accountS2 are allowed (in notation accountS1 → accountS2): 3 (negative) → 3 (negative) 3 (negative) → 2 (empty) 3 (negative) → 1 (low) 2 (empty) → 1 (low) 1 (low) → 1 (low) 1 (low) → 0 (ok) 0 (ok) → 0 (ok) IF (transition not allowed) THEN TP failed ENDIF
9			IF (accountS2 equals to 0 (ok)) THEN TP passed ELSE { accountS1 = accountS2 GOTO STEP 4 ENDIF
10		⇐	ChargeReportResponse = { chargeReportRespSender = any, dataReceived = ChargeReport.timeofReport ChargeReport.mileage ChargeReport.chargeReportCounter, versionsResponse = ∅, obeStatusForDriver = ∅, accountUpdate = ∅ } TP passed

Table A.41 — TP_BEFE_BV_28

TP_BEFE_BV_28	Verify that DUT updates Account Status for event based on-board accounts upon AddToAccount operation
TP origin	Specific
Reference	ISO 17575-1:2016, 7.7.5
Initial condition	Front End received already Context Data: — applicable for on-board accounts, — requesting presence of mileage or chargeReportCounter or timeofReport, and — requesting presence of accountStatus. Front End implementation supports on-board account. On-board account is event based. Threshold for on-board account is set to THR. The Front End shall have the value of accountStatus set other than “ok”. Front End is initialized and has a toll context activated. No accountable event happens during whole test purpose. Authentication is not used.

Table A.41 (continued)

Stimulus and expected behaviour			
	IUT		Tester
1	ChargeReport = { protocolVersion, obeld, vehicleLPNr, paymentMeans, serviceProviderContract, tollContext, chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, usageStatementList, sumVatForThisSession, accountStatus = accountS1, chargeReportCounter, mileage, listOfCCCAttributes }	⇒	
2			Verify the structure of sent ChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1 .
3			IF verify NOT "OK" THEN TP failed
4		⇐	ChargeReportResponse = { chargeReportRespSender = any, dataReceived = ChargeReport.timeofReport ChargeReport.mileage ChargeReport.chargeReportCounter, versionsResponse = ∅, obeStatusForDriver = ∅, accountUpdate = accountU } where accountU = addToAccount { addCredit = ∅, addDistance = ∅ addTime = ∅, addDuration = ∅, 0 < addEvents THR, }

Table A.41 (continued)

5	ChargeReport = { protocolVersion, obeld, vehicleLPNr, paymentMeans, serviceProviderContract, tollContext, chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, usageStatementList, sumVatForThisSession, accountStatus = accountS2, chargeReportCounter, mileage, listOfCCCAttributes }	⇒	
6			Verify the structure of sent ChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1 .
7			IF verify NOT "OK" THEN TP failed
8			The following transitions from accountS1 to accountS2 are allowed (in notation accountS1 → account S2): 3 (negative) → 3 (negative) 3 (negative) → 2 (empty) 3 (negative) → 1 (low) 2 (empty) → 1 (low) 1 (low) → 1 (low) 1 (low) → 0 (ok) 0 (ok) → 0 (ok) IF (transition not allowed) THEN TP failed ENDIF
9			IF (accountS2 equals to 0 (ok)) THEN TP passed ELSE { accountS1 = accountS2 GOTO STEP 4 ENDIF
10		⇐	ChargeReportResponse = { chargeReportRespSender = any, dataReceived = ChargeReport.timeofReport ChargeReport.mileage ChargeReport.chargeReportCounter, versionsResponse = ∅, obeStatusForDriver = ∅, accountUpdate = ∅ } TP passed

Table A.42 — TP_BEF_FE_BV_29

TP_BEF_FE_BV_29	Verify that DUT updates Account Status for credit based on-board accounts upon ReloadAccount operation		
TP origin	Specific		
Reference	ISO 17575-1:2016, 7.7.3		
Initial condition	Front End received already Context Data: — applicable for on-board accounts, — requesting presence of mileage or chargeReportCounter or timeofReport, and — requesting presence of accountStatus. Front End implementation supports on-board account. On-board account is credit based. Threshold for on-board account is set to THR. Predefined value for reload is less than THR/2, but greater than 0. The Front End shall have the value of accountStatus set other than “ok”. Front End is initialized and has a toll context activated. Front End shall not utilize any credits during the whole test purpose. Authentication is not used.		
Stimulus and expected behaviour			
	IUT		Tester
1	ChargeReport = { protocolVersion, obeld, vehicleLPNr, paymentMeans, serviceProviderContract, tollContext, chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, usageStatementList, sumVatForThisSession, accountStatus = accountS1, chargeReportCounter, mileage, listOfCCCAttributes	⇒	
2			Verify the structure of sent ChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1 .
3			IF verify NOT “OK” THEN TP failed

Table A.42 (continued)

4		⇐	ChargeReportResponse = { chargeReportRespSender = any, dataReceived = ChargeReport.timeofReport ChargeReport.mileage ChargeReport.chargeReportCounter, versionsResponse = ∅, obeStatusForDriver = ∅, accountUpdate = accountU } where accountU = reloadAccount { reloadOldCreditAmount = T, reloadOldDistanceLimit = ∅, reloadOldTimeLimit = ∅, reloadOldDurationLimit = ∅, reloadOldEventLimit = ∅, }
5	ChargeReport = { protocolVersion, obeld, vehicleLPNr, paymentMeans, serviceProviderContract, tollContext, chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, usageStatementList, sumVatForThisSession, accountStatus = accountS2, chargeReportCounter, mileage, listOfCCCAttributes }	⇒	
6			Verify the structure of sent ChargeReport, taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1.

Table A.42 (continued)

7			IF verify NOT "OK" THEN TP failed
8			The following transitions from accountS1 to accountS2 are allowed (in notation accountS1 → account S2): 3 (negative) → 3 (negative) 3 (negative) → 2 (empty) 3 (negative) → 1 (low) 2 (empty) → 1 (low) 1 (low) → 1 (low) 1 (low) → 0 (ok) 0 (ok) → 0 (ok) IF (transition not allowed) THEN TP failed ENDIF
9		⇐	ChargeReportResponse = { chargeReportRespSender = any, dataReceived = ChargeReport.timeofReport ChargeReport.mileage ChargeReport.chargeReportCounter, versionsResponse = ∅, obeStatusForDriver = ∅, accountUpdate = ∅ } TP passed

Table A.43 — TP_BEFE_BV_30

TP_BEFE_BV_30	Verify that DUT updates Account Status for distance based on-board accounts upon ReloadAccount operation
TP origin	Specific
Reference	ISO 17575-1:2016, 7.7.3
Initial condition	Front End received already Context Data: — applicable for on-board accounts, — requesting presence of mileage or chargeReportCounter or timeofReport, and — requesting presence of accountStatus. Front End implementation supports on-board account. On-board account is distance based. Threshold for on-board account is set to THR. Predefined value for reload is less than THR/2, but greater than 0. The Front End shall have the value of accountStatus set other than "ok". Front End is initialized and has a toll context activated. Front End shall not change its mileage during the whole test purpose Authentication is not used.

Table A.43 (continued)

Stimulus and expected behaviour			
	IUT		Tester
1	ChargeReport = { protocolVersion, obeId, vehicleLPNr, paymentMeans, serviceProviderContract, tollContext, chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, usageStatementList, sumVatForThisSession, accountStatus = accountS1, chargeReportCounter, mileage, listOfCCCAttributes }	⇒	
2			Verify the structure of sent ChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1 .
3			IF verify NOT "OK" THEN TP failed
4		⇐	ChargeReportResponse = { chargeReportRespSender = any, dataReceived = ChargeReport.timeofReport ChargeReport.mileage ChargeReport.chargeReportCounter, versionsResponse = ∅, obeStatusForDriver = ∅, accountUpdate = accountU } where accountU = reloadAccount { reloadOldCreditAmount = ∅, reloadOldDistanceLimit = T, reloadOldTimeLimit = ∅, reloadOldDurationLimit = ∅, reloadOldEventLimit = ∅, }

Table A.43 (continued)

5	ChargeReport = { protocolVersion, obeld, vehicleLPNr, paymentMeans, serviceProviderContract, tollContext, chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, usageStatementList, sumVatForThisSession, accountStatus = accountS2, chargeReportCounter, mileage, listOfCCCAttributes }	⇒	
6			Verify the structure of sent ChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1 .
7			IF verify NOT "OK" THEN TP failed
8			The following transitions from accountS1 to accountS2 are allowed (in notation accountS1 → account S2): 3 (negative) → 3 (negative) 3 (negative) → 2 (empty) 3 (negative) → 1 (low) 2 (empty) → 1 (low) 1 (low) → 1 (low) 1 (low) → 0 (ok) 0 (ok) → 0 (ok) IF (transition not allowed) THEN TP failed ENDIF
9		⇐	ChargeReportResponse = { chargeReportRespSender = any, dataReceived = ChargeReport.timeofReport ChargeReport.mileage ChargeReport.chargeReportCounter, versionsResponse = ∅, obeStatusForDriver = ∅, accountUpdate = ∅ } TP passed

Table A.44 — TP_BEFE_BV_31

TP_BEFE_BV_31	Verify that DUT updates Account Status for event based on-board accounts upon ReloadAccount operation		
TP origin	Specific		
Reference	ISO 17575-1:2016, 7.7.3		
Initial condition	Front End received already Context Data: — applicable for on-board accounts, — requesting presence of mileage or chargeReportCounter or timeofReport, and — requesting presence of accountStatus. Front End implementation supports on-board account. On-board account is event based. Threshold for on-board account is set to THR. Predefined value for reload is less than THR/2, but greater than 0. Front End is initialized and has a toll context activated. No accountable event happens during whole test purpose. Authentication is not used.		
Stimulus and expected behaviour			
	IUT		Tester
1	ChargeReport = { protocolVersion, obeld, vehicleLPNr, paymentMeans, serviceProviderContract, tollContext, chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, usageStatementList, sumVatForThisSession, accountStatus = accountS1, chargeReportCounter, mileage, listOfCCCAttributes }	⇒	
2			Verify the structure of sent ChargeReport , taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1 .
3			IF verify NOT “OK” THEN TP failed

Table A.44 (continued)

4		⇐	ChargeReportResponse = { chargeReportRespSender = any, dataReceived = ChargeReport.timeofReport ChargeReport.mileage ChargeReport.chargeReportCounter, versionsResponse = ∅, obeStatusForDriver = ∅, accountUpdate = accountU } where accountU = reloadAccount { reloadOldCreditAmount = ∅, reloadOldDistanceLimit = ∅, reloadOldTimeLimit = ∅, reloadOldDurationLimit = ∅, reloadOldEventLimit = T, }
5	ChargeReport = { protocolVersion, obeId, vehicleLPNr, paymentMeans, serviceProviderContract, tollContext, chargeReportFinalRecipient, timeOfReport, reportPeriod, versionInfo, usageStatementList, sumVatForThisSession, accountStatus = accountS2, chargeReportCounter, mileage, listOfCCCAttributes }	⇒	
6			Verify the structure of sent ChargeReport, taking presence and absence of data elements into account and verify allowed values of present data elements according to Table C.1.

Table A.44 (continued)

7			IF verify NOT "OK" THEN TP failed
8			The following transitions from accountS1 to accountS2 are allowed (in notation accountS1 → account S2): 3 (negative) → 3 (negative) 3 (negative) → 2 (empty) 3 (negative) → 1 (low) 2 (empty) → 1 (low) 1 (low) → 1 (low) 1 (low) → 0 (ok) 0 (ok) → 0 (ok) IF (transition not allowed) THEN TP failed ENDIF
9		⇐	ChargeReportResponse = { chargeReportRespSender = any, dataReceived = ChargeReport.timeofReport ChargeReport.mileage ChargeReport.chargeReportCounter, versionsResponse = ∅, obeStatusForDriver = ∅, accountUpdate = ∅ } TP passed

Table A.45 — TP_BEF_FE_BV_32

TP_BEF_FE_BV_32	Verify that DUT does not update Account Status for credit based on-board accounts upon ReloadAccount operation with "false" value
TP origin	Specific
Reference	ISO 17575-1:2016, 7.7.3
Initial condition	Front End received already Context Data: — applicable for on-board accounts, — requesting presence of mileage or chargeReportCounter or timeofReport, and — requesting presence of accountStatus. Front End implementation supports on-board account. On-board account is credit based. Threshold for on-board account is set to THR. Predefined value for reload is greater than THR. Front End credit based on-board account status is "low". Front End is initialized and has a toll context activated. Front End shall not utilize any credits during the whole test purpose. Authentication is not used.