TECHNICAL SPECIFICATION

ISO/TS 16407-1

First edition
2011-10-15

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

Part 1: Test suite structure and test purposes

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

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

Reference number
ISO/TS 16407-1:2011(E)

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

Published in Switzerland

Contents

Page

Foreword

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

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

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

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

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

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

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order to decide whether it will be confirmed for a further three years, revised to become an International Standard, or withdrawn.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

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

ISO/TS 16407-1 was prepared by Technical Committee ISO/TC 204, *Intelligent transport systems*, in collaboration with Technical Committee CEN/TC 278, *Road Transport and Traffic Telematics*.

ISO/TS 16407 consists of the following parts, under the general title *Electronic fee collection — Evaluation of equipment for conformity to ISO/TS 17575-1*:

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

— *Part 2: Abstract test suite*

Introduction

This part of ISO/TS 16407 is part of a set of standards that supports interoperability of autonomous electronic fee collection (EFC) systems, which includes ISO/TS 17575 parts 1 to 4 that define the EFC context data, their charge reports and their use of communication infrastructure.

Within the suite of EFC standards, this conformance evaluation procedure defines the process and tests for conformity evaluation of Front End and Back End that comply with the requirements in ISO/TS 17575-1.

This part of ISO/TS 16407 is intended to

— assess Front End and Back End capabilities,

— assess Front End and Back End behaviour,

— serve as a guide for Front End and Back End conformance evaluations and type approvals,

— achieve comparability between the results of the corresponding tests applied in different places at different times, and

— facilitate communication between parties.

This part of ISO 16407 is based on

— ISO/TS 17575-1, and

— the ISO 9646 family of standards on conformance test methodology.

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

Part 1:
Test suite structure and test purposes

1 Scope

This part of ISO/TS 16407 specifies the test suite structure (TSS) and test purposes (TP) to evaluate the conformity of Front End and Back End to ISO/TS 17575-1.

The objective of this part of ISO/TS 16407 is to provide a basis for conformance tests for the Front End and the Back End in electronic fee collection (EFC) based on autonomous on-board equipment (OBE) to enable interoperability between different equipment supplied by different manufacturers.

Autonomous OBE operates without relying on dedicated road-side infrastructure by employing wide-area technologies such as global navigation satellite systems (GNSS) and cellular communications networks (CN). These EFC systems are referred to by a variety of names. Besides the terms autonomous systems and GNSS/CN systems, also the terms GPS/GSM systems and wide-area charging systems are in use.

Autonomous systems use satellite positioning, often combined with additional sensor technologies such as gyroscopes, odometers, and accelerometers, to localise 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.

The testing of the following behaviours and functionalities is outside of the scope of this part of ISO/TS 16407:

— dynamic behaviour, i.e. sequence of messages and triggering events that must be exchanged/happen to fulfil certain charging scenarios;

— profiles and business logic built on top of particular pricing schemas;

— authentication, as its handling is not described in ISO/TS 17575-1;

— account update procedure ("reload" and "add to account") with respect to time and duration based on on-board accounts, as run-time environment has significant impact on test purpose outcome.

As ISO/TS 17575-1 does not specify any invalid behaviour of Front End and Back End, BI test purposes are not applicable for any test purpose group.

As ISO/TS 17575-1 does not define which of the data elements shall be present in the charge report response (CRR) (and under which conditions), the scope of test purposes (TP) for Back End is very limited.

2 Normative references

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

ISO/IEC 9646-6, *Information technology — Open Systems Interconnection — Conformance testing methodology and framework — Part 6: Protocol profile test specification*

ISO/TS 17575-1, *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.

3.1
attribute
application information formed by one or by a sequence of data elements, and that is managed by different actions used for implementation of a transaction

[ISO 14906:2011, definition 3.3]

3.2
authenticator
data appended to, or a cryptographic transformation of, a data unit that allows a recipient of the data unit to prove the source and/or the integrity of the data unit and protect against forgery

[ISO 14906:2011, definition 3.4]

3.3
Back End
generic name for the computing and communication facilities of the Service Provider and/or the Toll Charger

[ISO/TS 17575-1:2010, definition 3.4]

3.4
charge report
data structure transmitted from the Front End to the Back End to report road usage data and supplementary related information

[ISO/TS 17575-1:2010, definition 3.5]

3.5
contract
expression of an agreement between two or more parties concerning the use of the road infrastructure

[ISO 14906:2011, definition 3.7]

3.6
data element
datum, which might itself consist of lower level data elements

[ISO/TS 17575-1:2010, definition 3.10]

3.7
Front End
part(s) of the toll system where road usage data for an individual road user are collected, processed and delivered to the Back End

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

[ISO/TS 17575-1:2010, definition 3.13]

3.8
service provider
operator that accepts the user's payment means and in return provides a road-use service to the user

NOTE Taken from ISO 14906:2004.

3.9
toll charger
legal entity charging a toll for vehicles in a toll domain

[ISO/TS 17574:2009, definition 3.27]

3.10
toll context
logical view of a toll scheme as defined by attributes and functions

[ISO/TS 17575-1:2010, definition 3.22]

3.11
toll regime
set of rules, including enforcement rules, governing the collection of toll in a toll

[ISO/TS 17575-1:2010, definition 3.25]

4 Abbreviated terms

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

ADU Application data unit (ISO/TS 17575-1)

ASN.1 Abstract Syntax Notation One (ISO/IEC 8824-1:2002)

ATS Abstract Test Suite

BI Invalid Behaviour

BV Valid Behaviour

CCC Compliance Check Communication (ISO/TS 12813)

CN Cellular network (ISO/TS 17575-1)

CRR Charge Report Response

DUT Device Under Test

EFC Electronic Fee Collection (ISO 17573)

GNSS Global Navigation Satellite Systems (ISO/TS 17575-1)

HMI	Human Machine Interface (ISO/TS 17575-1)
ID	Identifier
OBE	On-board Equipment (ISO/TS 17575-1)
PCTR	Proforma Conformance Test Report
PICS	Protocol Implementation Conformance Statements
TP	Test Purposes
TSS	Test Suite Structure
VAT	Value Added Tax (ISO/TS 17575-1)

5 Value added tax (VAT) Test Suite Structure (TSS)

5.1 Structure

Table 1 shows the Test Suite Structure (TSS).

Table 1 — Test Suite Structures

Group	Type of DUT	Behaviour
Charge Report	Front End	Valid Behaviour
		Invalid Behaviour not applicable
Back End Feedback	Front End	Valid Behaviour
		Invalid Behaviour not applicable
Charge Report Response	Back End	Valid Behaviour
		Invalid Behaviour 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

a) for test purposes that are **identical** to those defined in 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;

b) 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 has to be modified for the profile conformance testing;

c) for test purposes that are **specific** to ISO/TS 17575-1, a complete description is given;

d) 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 below. 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/TS 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 the sub-clause below.
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 DUT 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 DUT 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>/<dut>/<x>-<nn>

TP : to indicate that it is a Test Purpose;

<group> : which group TP belongs to;

<dut> : type of DUT (i.e. FE or BE);

X : type of testing (i.e. Valid Behaviour tests – BV, or Invalid Behaviour 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>/<dut>/<x>-<nn>		
<group>		
applicable for FE	CR	Charge Report
applicable for FE	BEF	Back End Feedback
applicable for BE	CRR	Charge Report Response
<dut> = type of DUT	FE	Front End
	BE	Back End
x = Type of testing	BV	Valid Behaviour Tests
	BI	Invalid Behaviour 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 Introduction

This annex contains the Test Purposes (TP) for the conformity evaluation of Front End to ISO/TS 17575-1.

A.1.1 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 below.

Table A.1 — Description of TP Symbols

SYMBOL	DESCRIPTION
XXX.rq ⇒	The Tester sends the XXX.rq to the DUT
⇐ YYY.rs	The DUT sends the YYY.rs to the Tester
⇐ YYY.rs = {attribute1, attribute2, attribute3}	The DUT 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.
⇐ YYY.rs = {attribute1= value1}	The DUT 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 = B	A “is equal to” B
A → B	A “is transformed” into B
Ø	Means “empty” or “not set”
A | 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, it has to be noted that the sequence of ADUs issued by an Front End is not constrained by ISO/TS 17575-1. This means that ADU cannot in general be forced to be generated by the DUT. 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 regime. Such situation is illustrated in Figure A.1.

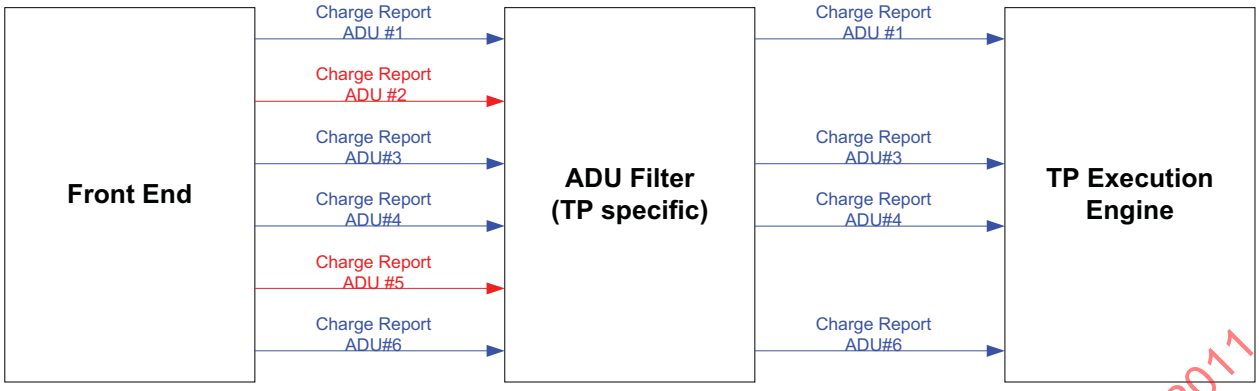

Figure A.1 — Handling of ADUs applicable for particular TP

A.2 Charge Report

These Test Purposes apply to data elements in Charge Report as claimed in ISO/TS 17575-1 Clause B.2/ChargeReport, Clause B.3.1, Clause B.3.3.

NOTE No test purposes for invalid behaviour are specified (BI), as ISO/TS 17575-1 does not specify any invalid behaviour of Front End.

A.2.1 BV test purposes

Test subgroup objective:

— to test the behaviour of the DUT with respect to data elements contained in Charge Report.

TP/CR/FE/BV/02	**Verify that Time of Report is according to the local time**
TP Origin	Specific
Reference	ISO/TS 17575-1, Clause 6.3.2
Initial Condition	Front End received already Context Data with correctly set timeZone attribute. Front End is initialized and has a toll context activated. No authentication is required.

Stimulus and Expected Behaviour

	DUT		Tester
1	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus, transactionCounter, mileage, listOfCCCAttributes, authenticator}	⇒	
2			Verify structure of sent **ChargeReport**, taking presence and absence of optional 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” OR timeofReport not present THEN TP passed ELSE TP failed ENDIF
6		⇐	**ChargeReportResponse** = { reportRecipientId = any, dataReceived = (ChargeReport.timeOfReport | ChargeReport.mileage | ChargeReport.transactionCounter), versionsResponse = ø, obeStatusForDriver = 0, accountUpdate = ø, responseAuthenticator = ø}

TP/CR/FE/BV/03	Verify that Transaction Counter is incremented
TP Origin	Specific
Reference	ISO/TS 17575-1, Clause 6.3.5
Initial Condition	Front End received already Context Data with Charge Report Configuration requesting transactionCounter to be present. Front End is initialized and has a toll context activated. No authentication is required.

Stimulus and Expected Behaviour

	DUT		Tester
1	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus, transactionCounter = v1, mileage, listOfCCCAttributes, authenticator}	⇒	
2			Verify structure of sent **ChargeReport**, taking presence and absence of optional 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** = { reportRecipientId = any, dataReceived = (ChargeReport.timeOfReport | ChargeReport.mileage | ChargeReport.transactionCounter), versionsResponse = ø, obeStatusForDriver = 0, accountUpdate = ø, responseAuthenticator = ø}
5	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus, transactionCounter = v2, mileage, listOfCCCAttributes, authenticator}	⇒	
6			Verify structure of sent **ChargeReport**, taking presence and absence of optional 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 equals to 1 THEN TP passed ELSE TP failed ENDIF
9		⇐	**ChargeReportResponse** = { reportRecipientId = any, dataReceived = (ChargeReport.timeOfReport | ChargeReport.mileage | ChargeReport.transactionCounter), versionsResponse = ø, obeStatusForDriver = 0, accountUpdate = ø, responseAuthenticator = ø}

TP/CR/FE/BV/04	Transaction Counter Overflow
TP Origin	Specific
Reference	ISO/TS 17575-1, Clause 6.3.5
Initial Condition	Front End received already Context Data with Charge Report Configuration requesting transactionCounter to be present. Front End is initialized and has a toll context activated. No authentication is required.

Stimulus and Expected Behaviour

	DUT		Tester
1	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus, transactionCounter = 4294967295, mileage, listOfCCCAttributes, authenticator}	⇒	
2			Verify structure of sent **ChargeReport**, taking presence and absence of optional 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** = { reportRecipientId = any, dataReceived = (ChargeReport.timeOfReport | ChargeReport.mileage | ChargeReport.transactionCounter), versionsResponse = ø, obeStatusForDriver = 0, accountUpdate = ø, responseAuthenticator = ø}
5	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus, transactionCounter = v1, mileage, listOfCCCAttributes, authenticator}	⇒	
6			Verify structure of sent **ChargeReport**, taking presence and absence of optional 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** = { reportRecipientId = any, dataReceived = (ChargeReport.timeOfReport | ChargeReport.mileage | ChargeReport.transactionCounter), versionsResponse = ø, obeStatusForDriver = 0, accountUpdate = ø, responseAuthenticator = ø}

TP/CR/FE/BV/05	**Verify that mileage is not decreasing within one contract**
TP Origin	Specific
Reference	ISO/TS 17575-1, Clause 6.3.6
Initial Condition	Front End received already Context Data with Charge Report Configuration requesting mileage to be present. Front End is initialized and has a toll context activated. No authentication is required.

Stimulus and Expected Behaviour

	DUT		Tester
1	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract = contract1, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus, transactionCounter, mileage = v1, listOfCCCAttributes, authenticator}	⇒	
2			Verify structure of sent **ChargeReport**, taking presence and absence of optional 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** = { reportRecipientId = any, dataReceived = (ChargeReport.timeOfReport | ChargeReport.mileage | ChargeReport.transactionCounter), versionsResponse = ø, obeStatusForDriver = 0, accountUpdate = ø, responseAuthenticator = ø}
5	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract = contract1, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus, transactionCounter, mileage = v2, listOfCCCAttributes, authenticator}	⇒	
6			Verify structure of sent **ChargeReport**, taking presence and absence of optional 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.dist >= v1.dist) AND (v2.disUnit equals to v1.disUnit)
			THEN TP passed
			ELSE TP failed
			ENDIF

9		⇐	**ChargeReportResponse** = { reportRecipientId = any, dataReceived = (ChargeReport.timeOfReport | ChargeReport.mileage | ChargeReport.transactionCounter), versionsResponse = ø, obeStatusForDriver = 0, accountUpdate = ø, responseAuthenticator = ø}

TP/CR/FE/BV/06	Verify that mileage rolls over
TP Origin	Specific
Reference	ISO/TS 17575-1, Clause 6.3.6
Initial Condition	Front End received already Context Data with Charge Report Configuration requesting mileage to be present. Front End is initialized and has a toll context activated. Mileage is about to roll-over, e.g. its value is close to 16777215 No authentication is required.

Stimulus and Expected Behaviour

	DUT		Tester
1	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract = contract1, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus, transactionCounter, mileage → 16777215^{-}, listOfCCCAttributes, authenticator}	⇒	
2			Verify structure of sent **ChargeReport**, taking presence and absence of optional 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** = { reportRecipientId = any, dataReceived = (ChargeReport.timeOfReport | ChargeReport.mileage | ChargeReport.transactionCounter), versionsResponse = ø, obeStatusForDriver = 0, accountUpdate = ø, responseAuthenticator = ø}
5	Mileage has changed enough to perform roll-over		
6	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract = contract1, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus, transactionCounter, mileage = v1, listOfCCCAttributes, authenticator}	⇒	
7			Verify structure of sent **ChargeReport**, taking presence and absence of optional 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^{+} THEN TP passed ELSE TP failed ENDIF
10		⇐	**ChargeReportResponse** = { reportRecipientId = any, dataReceived = (ChargeReport.timeOfReport | ChargeReport.mileage | ChargeReport.transactionCounter), versionsResponse = ø, obeStatusForDriver = 0, accountUpdate = ø, responseAuthenticator = ø}

TP/CR/FE/BV/07	Verify that OBE ID is fixed
TP Origin	Specific
Reference	ISO/TS 17575-1, Clause 6.4.1
Initial Condition	Front End received already Context Data with Charge Report Configuration requesting obeId to be present. Front End is initialized and has a toll context activated. No authentication is required.

Stimulus and Expected Behaviour

	DUT		Tester
1	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus, transactionCounter, mileage, listOfCCCAttributes, authenticator}	⇒	
2			Verify structure of sent **ChargeReport**, taking presence and absence of optional 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** = { reportRecipientId = any, dataReceived = (ChargeReport.timeOfReport | ChargeReport.mileage | ChargeReport.transactionCounter), versionsResponse = ø, obeStatusForDriver = 0, accountUpdate = ø, responseAuthenticator = ø}
5	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus, transactionCounter, mileage, listOfCCCAttributes, authenticator}	⇒	
6			Verify structure of sent **ChargeReport**, taking presence and absence of optional 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** = { reportRecipientId = any, dataReceived = (ChargeReport.timeOfReport | ChargeReport.mileage | ChargeReport.transactionCounter), versionsResponse = ø, obeStatusForDriver = 0, accountUpdate = ø, responseAuthenticator = ø}
...			
n	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus, transactionCounter, mileage, listOfCCCAttributes, authenticator}	⇒	

n+1		⇐	**ChargeReportResponse** = { reportRecipientId = any, dataReceived = (ChargeReport.timeOfReport | ChargeReport.mileage | ChargeReport.transactionCounter), versionsResponse = ø, obeStatusForDriver = 0, accountUpdate = ø, responseAuthenticator = ø}
n+2			Verify structure of sent **ChargeReport**, taking presence and absence of optional data elements into account and verify allowed values of present data elements according to Table C.1
n+3			IF verify NOT "OK" THEN TP failed

n+4			IF (obeId the same in every **ChargeReport**)
			THEN TP passed
			ELSE TP failed
			ENDIF

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/TS 17575-1, Clause 6.4.2
Initial Condition	Front End received already Context Data. Front End is initialized and has a toll context activated. No authentication is required.

Stimulus and Expected Behaviour

	DUT		Tester
1	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus, transactionCounter, mileage, listOfCCCAttributes, authenticator}	⇒	
2			Verify structure of sent **ChargeReport**, taking presence and absence of optional 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			IF (vehicleLPNr present in usageStatementList at any level) AND (vehicleLPNr present in **ChargeReport**)
			THEN TP failed
			ELSE TP passed
			ENDIF

5		⇐	**ChargeReportResponse** = { reportRecipientId = any, dataReceived = (ChargeReport.timeOfReport | ChargeReport.mileage | ChargeReport.transactionCounter), versionsResponse = ø, obeStatusForDriver = 0, accountUpdate = ø, responseAuthenticator = ø}

TP/CR/FE/BV/09	**Verify the uniqueness of Usage Statement ID**
TP Origin	Specific
Reference	ISO/TS 17575-1, Clause 6.5.2
Initial Condition	Front End received already Context Data. Front End is initialized and has a toll context activated. No authentication is required.

Stimulus and Expected Behaviour

	DUT		Tester
1	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract = contract1, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus, transactionCounter, mileage, listOfCCCAttributes, authenticator}	⇒	
2			Verify structure of sent **ChargeReport**, taking presence and absence of optional 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			IF (usageStatementList present in **ChargeReport**) AND NOT (every usageStatementId unique within usageStatementList) THEN TP failed ELSE TP passed ENDIF
5		⇐	**ChargeReportResponse** = { reportRecipientId = any, dataReceived = (ChargeReport.timeOfReport | ChargeReport.mileage | ChargeReport.transactionCounter), versionsResponse = ø, obeStatusForDriver = 0, accountUpdate = ø, responseAuthenticator = ø}

TP/CR/FE/BV/10	**Verify that begin period is lower than end period in Report Period**
TP Origin	Specific
Reference	ISO/TS 17575-1, Clause 6.3.3
Initial Condition	Front End received already Context Data with Charge Report Configuration requesting reportPeriod to be present. Front End is initialized and has a toll context activated. No authentication is required.

Stimulus and Expected Behaviour

	DUT		Tester
1	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract = contract1, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus, transactionCounter, mileage, listOfCCCAttributes, authenticator}	⇒	
2			Verify structure of sent **ChargeReport**, taking presence and absence of optional 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			IF (reportPeriod.beginOfPeriod < reportPeriod.endOfPeriod) THEN TP passed ELSE TP failed ENDIF
5		⇐	**ChargeReportResponse** = { reportRecipientId = any, dataReceived = (ChargeReport.timeOfReport | ChargeReport.mileage | ChargeReport.transactionCounter), versionsResponse = ø, obeStatusForDriver = 0, accountUpdate = ø, responseAuthenticator = ø}

TP/CR/FE/BV/11	**Verify that begin period is lower than end period in Aggregated Fee**
TP Origin	Specific
Reference	ISO/TS 17575-1, Clause 6.5.4
Initial Condition	Front End received already Context Data with Charge Report Configuration requesting aggregatedFee to be present. Front End is initialized and has a toll context activated. No authentication is required.

Stimulus and Expected Behaviour

	DUT		Tester
1	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract = contract1, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus, transactionCounter, mileage, listOfCCCAttributes, authenticator}	⇒	
2			Verify structure of sent **ChargeReport**, taking presence and absence of optional 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			IF each usageStatementList[i].aggregatedFee.timePeriodCovered.beginOfPeriod < usageStatementList[i].aggregatedFee.timePeriodCovered.endOfPeriod (where i=1..number usageStatement) THEN TP passed ELSE TP failed ENDIF

TP/CR/FE/BV/12	**Verify that begin period is lower than end period in Aggregated Single Tariff Class Session**
TP Origin	Specific
Reference	ISO/TS 17575-1, Clause 6.5.5
Initial Condition	Front End received already Context Data with Charge Report Configuration requesting timePeriodCovered (associated with aggregatedSingleClassTariffSession) to be present. Front End is initialized and has a toll context activated. No authentication is required.

Stimulus and Expected Behaviour

	DUT		Tester
1	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract = contract1, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus, transactionCounter, mileage, listOfCCCAttributes, authenticator}	⇒	
2			Verify structure of sent **ChargeReport**, taking presence and absence of optional 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			IF each usageStatementList[i]. aggregatedSingleTariffClassSession.timePeriodCovered.beginOfPeriod < usageStatementList[i]. aggregatedSingleTariffClassSession.timePeriodCovered.endOfPeriod (where i=1..number usageStatement) THEN TP passed ELSE TP failed ENDIF
5		⇐	**ChargeReportResponse** = { reportRecipientId = any, dataReceived = (ChargeReport.timeOfReport | ChargeReport.mileage | ChargeReport.transactionCounter), versionsResponse = ø, obeStatusForDriver = 0, accountUpdate = ø, responseAuthenticator = ø}

TP/CR/FE/BV/13	**Verify that reported Tariff Class is defined in Context Data (Aggregated Single Class Tariff Session)**
TP Origin	Specific
Reference	ISO/TS 17575-1, Clause 6.5.6
Initial Condition	Front End received already Context Data with Charge Report Configuration requesting tariffClass (associated with aggregatedSingleClassTariffSession) to be present. Front End is initialized and has a toll context activated. No authentication is required.

Stimulus and Expected Behaviour

	DUT		Tester
1	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract = contract1, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus, transactionCounter, mileage, listOfCCCAttributes, authenticator}	⇒	
2			Verify structure of sent **ChargeReport**, taking presence and absence of optional 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			Condition#1. Verify whether each usageStatementList[i].aggregatedSingleTariffClassSession.tariffClass.locationClassId defined in Context Data Condition#2. Verify whether each usageStatementList[i].aggregatedSingleTariffClassSession.tariffClass.timeClassId defined in Context Data Condition#3. Verify whether each usageStatementList[i].aggregatedSingleTariffClassSession.tariffClass.userClassId defined in Context Data where i=1..number usageStatement

5			IF (Condition#1 OK) AND (Condition#2 OK) AND (Condition#3 OK)
			THEN TP passed
			ELSE TP failed
			ENDIF

TP/CR/FE/BV/14	**Verify that reported Tariff Class is defined in Context Data (List of Charged Objects)**
TP Origin	Specific
Reference	ISO/TS 17575-1, Clause 6.5.7
Initial Condition	Front End received already Context Data with Charge Report Configuration requesting tariffClass (associated with DetectedChargeObjectContent) to be present. Front End is initialized and has a toll context activated. No authentication is required.

Stimulus and Expected Behaviour

	DUT		Tester
1	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract = contract1, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus, transactionCounter, mileage, listOfCCCAttributes, authenticator}	⇒	
2			Verify structure of sent **ChargeReport**, taking presence and absence of optional 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			Condition#1. Verify whether each usageStatementList[i].listOfChargeObjects[j].tariffClass.locationClassId defined in Context Data Condition#2. Verify whether each usageStatementList[i].listOfChargeObjects[j].tariffClass.timeClassId defined in Context Data Condition#3. Verify whether each usageStatementList[i].listOfChargeObjects[j].tariffClass.userClassId defined in Context Data where i=1..number usageStatement j = 1.. number listOfChargedObjets for particular usageStatement

5			IF (Condition#1 OK) AND (Condition#2 OK) AND (Condition#3 OK)
			THEN TP passed
			ELSE TP failed
			ENDIF

6		⇐	**ChargeReportResponse** = { reportRecipientId = any, dataReceived = (ChargeReport.timeOfReport | ChargeReport.mileage | ChargeReport.transactionCounter), versionsResponse = ø, obeStatusForDriver = 0, accountUpdate = ø, responseAuthenticator = ø}

TP/CR/FE/BV/15	**Verify that reported Tariff Class is defined in Context Data (List of Raw Usage Data)**
TP Origin	Specific
Reference	ISO/TS 17575-1, Clause 6.5.9
Initial Condition	Front End received already Context Data with Charge Report Configuration requesting tariffClass (associated with ListOfRawUsageDataContent) to be present. Front End is initialized and has a toll context activated. No authentication is required.

Stimulus and Expected Behaviour

	DUT		Tester
1	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract = contract1, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus, transactionCounter, mileage, listOfCCCAttributes, authenticator}	⇒	
2			Verify structure of sent **ChargeReport**, taking presence and absence of optional 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			Condition#1. Verify whether each usageStatementList[i].listOfRawUsageData.tariffClass.locationClassId defined in Context Data Condition#2. Verify whether each usageStatementList[i].listOfRawUsageData.tariffClass.timeClassId defined in Context Data Condition#3. Verify whether each usageStatementList[i].listOfRawUsageData.tariffClass.userClassId defined in Context Data where i=1..number usageStatement

5			IF (Condition#1 OK) AND (Condition#2 OK) AND (Condition#3 OK)
			THEN TP passed
			ELSE TP failed
			ENDIF

6		⇐	**ChargeReportResponse** = { reportRecipientId = any, dataReceived = (ChargeReport.timeOfReport | ChargeReport.mileage | ChargeReport.transactionCounter), versionsResponse = ø, obeStatusForDriver = 0, accountUpdate = ø, responseAuthenticator = ø}

A.2.2 BI test purposes

No BI test purposes are applicable for this TP group.

A.3 Back End Feedback

These Test Purposes apply to Back End feedback sent in Charge Report Response as claimed in ISO/TS 17575-1 Clause B.2.ChargeReportResponse, Clause B.3.2, Clause B.3.4.

NOTE No test purposes for invalid behaviour are specified (BI), as ISO/TS 17575-1 does not specify any invalid behaviour of Front End.

A.3.1 BV test purposes

Test subgroup objective:

— to test DUT behaviour with respect to OBE Status for Driver;

— to test the DUT behaviour with respect account update handling for the following types of on-board accounts:

  — credit;

  — distance;

  — time;

  — duration;

  — event.

TP/BEF/FE/BV/01	**Verify that OBE Status For Driver is 'OK' having received Charge Report Response**
TP Origin	Specific
Reference	ISO/TS 17575-1, Clause 6.4.7
Initial Condition	Front End received already Context Data with Charge Report Configuration requesting transactionCounter or mileage to be present. Front End is initialized and has a toll context activated. No authentication is required.

Stimulus and Expected Behaviour

	DUT		Tester
1	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus, transactionCounter, mileage, listOfCCCAttributes, authenticator}	⇒	
2			Verify structure of sent **ChargeReport**, taking presence and absence of optional 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** = { reportRecipientId = any, dataReceived = (ChargeReport.timeOfReport | ChargeReport.mileage | ChargeReport.transactionCounter), versionsResponse = ø, obeStatusForDriver = 0, accountUpdate = ø, responseAuthenticator = ø}
5	Verify that OBE status 'OK'		
6	IF verify "OK" THEN TP passed ELSE TP failed ENDIF		

TP/BEF/FE/BV/02	**Verify that OBE Status For Driver is 'NOK' having received Charge Report Response**
TP Origin	Specific
Reference	ISO/TS 17575-1, Clause 6.4.7
Initial Condition	Front End received already Context Data with Charge Report Configuration requesting transactionCounter or mileage to be present. Front End is initialized and has a toll context activated. No authentication is required.

Stimulus and Expected Behaviour

	DUT		Tester
1	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus, transactionCounter, mileage, listOfCCCAttributes, authenticator}	⇒	
2			Verify structure of sent **ChargeReport**, taking presence and absence of optional 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** = { reportRecipientId = any, dataReceived = (ChargeReport.timeOfReport | ChargeReport.mileage | ChargeReport.transactionCounter), versionsResponse = ø, obeStatusForDriver = 1, accountUpdate = ø, responseAuthenticator = ø}
5	Verify that OBE status 'NOK'		
6	IF verify "OK" THEN TP passed ELSE TP failed ENDIF		

TP/BEF/FE/BV/03	**Verify that OBE Status For Driver is 'contactOperator' having received Charge Report Response**
TP Origin	Specific
Reference	ISO/TS 17575-1, Clause 6.4.7
Initial Condition	Front End received already Context Data with Charge Report Configuration requesting transactionCounter or mileage to be present. Front End is initialized and has a toll context activated. No authentication is required.

Stimulus and Expected Behaviour

	DUT		Tester
1	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus, transactionCounter, mileage, listOfCCCAttributes, authenticator}	⇒	
2			Verify structure of sent **ChargeReport**, taking presence and absence of optional 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** = { reportRecipientId = any, dataReceived = (ChargeReport.timeOfReport | ChargeReport.mileage | ChargeReport.transactionCounter), versionsResponse = ø, obeStatusForDriver = 2, accountUpdate = ø, responseAuthenticator = ø}
5	Verify that OBE status 'contactOperator'		
6	IF verify "OK" THEN TP passed ELSE TP failed ENDIF		

TP/BEF/FE/BV/04	**Verify that OBE Status For Driver is 'nokInLocalContext' having received Charge Report Response**
TP Origin	Specific
Reference	ISO/TS 17575-1, Clause 6.4.7
Initial Condition	Front End received already Context Data with Charge Report Configuration requesting transactionCounter or mileage to be present. Front End is initialized and has a toll context activated. No authentication is required.

Stimulus and Expected Behaviour

	DUT		Tester
1	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus, transactionCounter, mileage, listOfCCCAttributes, authenticator}	⇒	
2			Verify structure of sent **ChargeReport**, taking presence and absence of optional 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** = { reportRecipientId = any, dataReceived = (ChargeReport.timeOfReport | ChargeReport.mileage | ChargeReport.transactionCounter), versionsResponse = ø, obeStatusForDriver = 3, accountUpdate = ø, responseAuthenticator = ø}
5	Verify that OBE status 'nokInLocalContext'		
6	IF verify "OK" THEN TP passed ELSE TP failed ENDIF		

TP/BEF/FE/BV/05	**Verify that OBE Status For Driver is 'noSignalling' having received Charge Report Response**
TP Origin	Specific
Reference	ISO/TS 17575-1, Clause 6.4.7
Initial Condition	Front End received already Context Data with Charge Report Configuration requesting transactionCounter or mileage to be present. Front End is initialized and has a toll context activated. No authentication is required.

Stimulus and Expected Behaviour

	DUT		Tester
1	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus, transactionCounter, mileage, listOfCCCAttributes, authenticator}	⇒	
2			Verify structure of sent **ChargeReport**, taking presence and absence of optional 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** = { reportRecipientId = any, dataReceived = (ChargeReport.timeOfReport | ChargeReport.mileage | ChargeReport.transactionCounter), versionsResponse = ø, obeStatusForDriver = 255, accountUpdate = ø, responseAuthenticator = ø}
5	Verify that OBE status 'noSignalling'		
6	IF verify "OK" THEN TP passed ELSE TP failed ENDIF		

TP/BEF/FE/BV/06	**Verify that DUT sets Account Status to 'OK' for credit based on-board accounts**
TP Origin	Specific
Reference	ISO/TS 17575-1, Clause 6.6
Initial Condition	Front End received already Context Data applicable for on-board accounts and with Charge Report Configuration requesting accountStatus, transactionCounter and mileage to be present. 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. No authentication is required.

Stimulus and Expected Behaviour

	DUT		Tester
1	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract = contract1, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus = accountS1, transactionCounter, mileage, listOfCCCAttributes, authenticator}	⇒	
2			Verify structure of sent **ChargeReport**, taking presence and absence of optional 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** = {
			reportRecipientId = any,
			dataReceived = (ChargeReport.timeOfReport |
			ChargeReport.mileage |
			ChargeReport.transactionCounter),
			versionsResponse = ø,
			obeStatusForDriver = ø,
			accountUpdate = accountU,
			responseAuthenticator = ø
			}
			where accountU = setAccount {
			newCreditLimit = {
			paymentFeeAmount > THR,
			paymentFeeUnit = UNIT
			}
			newDistanceLimit = ø,
			newTimeLimit = ø,
			newDurationLimit = ø,
			newEventLimit = ø,
			newAuthenticator = ø }

5	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract = contract1, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus = accountS2, transactionCounter, mileage, listOfCCCAttributes, authenticator} No credit change shall occur before sending this data element		

6			IF (accountS2 equals to 0)
			THEN TP passed
			ELSE TP failed
			ENDIF

7		⇐	**ChargeReportResponse** = { reportRecipientId = any, dataReceived = (ChargeReport.timeOfReport | ChargeReport.mileage | ChargeReport.transactionCounter), versionsResponse = ø, obeStatusForDriver = 0, accountUpdate = ø, responseAuthenticator = ø}

TP/BEF/FE/BV/07	**Verify that DUT sets Account Status to 'LOW' for credit based on-board accounts**
TP Origin	Specific
Reference	ISO/TS 17575-1, Clause 6.6
Initial Condition	Front End received already Context Data applicable for on-board accounts and with Charge Report Configuration requesting accountStatus, transactionCounter and mileage to be present. 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. No authentication is required.

Stimulus and Expected Behaviour

	DUT		Tester
1	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract = contract1, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus = accountS1, transactionCounter, mileage, listOfCCCAttributes, authenticator}	⇒	
2			Verify structure of sent **ChargeReport**, taking presence and absence of optional 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** = {
			reportRecipientId = any,
			dataReceived = (ChargeReport.timeOfReport |
			ChargeReport.mileage |
			ChargeReport.transactionCounter),
			versionsResponse = ø,
			obeStatusForDriver = ø,
			accountUpdate = accountU,
			responseAuthenticator = ø
			}
			where accountU = setAccount {
			newCreditLimit = {
			0 < paymentFeeAmount < THR,
			paymentFeeUnit = UNIT
			}
			newDistanceLimit = ø,
			newTimeLimit = ø,
			newDurationLimit = ø,
			newEventLimit = ø,
			newAuthenticator = ø }

5	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract = contract1, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus = accountS2, transactionCounter, mileage, listOfCCCAttributes, authenticator} No credit change shall occur before sending this data element		

6			IF (accountS2 equals to 1)
			THEN TP passed
			ELSE TP failed
			ENDIF

7		⇐	**ChargeReportResponse** = { reportRecipientId = any, dataReceived = (ChargeReport.timeOfReport | ChargeReport.mileage | ChargeReport.transactionCounter), versionsResponse = ø, obeStatusForDriver = 0, accountUpdate = ø, responseAuthenticator = ø}

TP/BEF/FE/BV/08	**Verify that DUT sets Account Status to 'EMPTY' for credit based on-board accounts**
TP Origin	Specific
Reference	ISO/TS 17575-1, Clause 6.6
Initial Condition	Front End received already Context Data applicable for on-board accounts and with Charge Report Configuration requesting accountStatus, transactionCounter and mileage to be present. 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. No authentication is required.

Stimulus and Expected Behaviour

	DUT		Tester
1	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract = contract1, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus = accountS1, transactionCounter, mileage, listOfCCCAttributes, authenticator}	⇒	
2			Verify structure of sent **ChargeReport**, taking presence and absence of optional 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** = {
			reportRecipientId = any,
			dataReceived = (ChargeReport.timeOfReport |
			ChargeReport.mileage |
			ChargeReport.transactionCounter),
			versionsResponse = ø,
			obeStatusForDriver = ø,
			accountUpdate = accountU,
			responseAuthenticator = ø
			}
			where accountU = setAccount {
			newCreditLimit = {
			paymentFeeAmount = 0,
			paymentFeeUnit = UNIT
			}
			newDistanceLimit = ø,
			newTimeLimit = ø,
			newDurationLimit = ø,
			newEventLimit = ø,
			newAuthenticator = ø }

5	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract = contract1, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus = accountS2, transactionCounter, mileage, listOfCCCAttributes, authenticator} No credit change shall occur before sending this data element		

6			IF (accountS2 equals to 2)
			THEN TP passed
			ELSE TP failed
			ENDIF

7		⇐	**ChargeReportResponse** = { reportRecipientId = any, dataReceived = (ChargeReport.timeOfReport | ChargeReport.mileage | ChargeReport.transactionCounter), versionsResponse = ø, obeStatusForDriver = 0, accountUpdate = ø, responseAuthenticator = ø}

TP/BEF/FE/BV/09	**Verify that DUT sets Account Status to 'NEGATIVE' for credit based on-board accounts**
TP Origin	Specific
Reference	ISO/TS 17575-1, Clause 6.6
Initial Condition	Front End received already Context Data applicable for on-board accounts and with Charge Report Configuration requesting accountStatus, transactionCounter and mileage to be present. 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. No authentication is required.

Stimulus and Expected Behaviour

	DUT		Tester
1	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract = contract1, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus = accountS1, transactionCounter, mileage, listOfCCCAttributes, authenticator}	⇒	
2			Verify structure of sent **ChargeReport**, taking presence and absence of optional 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** = {
			reportRecipientId = any,
			dataReceived = (ChargeReport.timeOfReport |
			ChargeReport.mileage |
			ChargeReport.transactionCounter),
			versionsResponse = ø,
			obeStatusForDriver = ø,
			accountUpdate = accountU,
			responseAuthenticator = ø
			}
			where accountU = setAccount {
			newCreditLimit = {
			paymentFeeAmount < 0,
			paymentFeeUnit = UNIT
			}
			newDistanceLimit = ø,
			newTimeLimit = ø,
			newDurationLimit = ø,
			newEventLimit = ø,
			newAuthenticator = ø }

5	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract = contract1, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus = accountS2, transactionCounter, mileage, listOfCCCAttributes, authenticator} No credit change shall occur before sending this data element		

6			IF (accountS2 equals to 3)
			THEN TP passed
			ELSE TP failed
			ENDIF

7		⇐	**ChargeReportResponse** = { reportRecipientId = any, dataReceived = (ChargeReport.timeOfReport | ChargeReport.mileage | ChargeReport.transactionCounter), versionsResponse = ø, obeStatusForDriver = 0, accountUpdate = ø, responseAuthenticator = ø}

TP/BEF/FE/BV/10	**Verify that DUT sets Account Status to 'OK' for distance based on-board accounts**
TP Origin	Specific
Reference	ISO/TS 17575-1, Clause 6.6
Initial Condition	Front End received already Context Data applicable for on-board accounts and with Charge Report Configuration requesting accountStatus, transactionCounter and mileage to be present. 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. No authentication is required.

Stimulus and Expected Behaviour

	DUT		Tester
1	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract = contract1, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus = accountS1, transactionCounter, mileage, listOfCCCAttributes, authenticator}	⇒	
2			Verify structure of sent **ChargeReport**, taking presence and absence of optional 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** = {
			reportRecipientId = any,
			dataReceived = (ChargeReport.timeOfReport |
			ChargeReport.mileage |
			ChargeReport.transactionCounter),
			versionsResponse = ø,
			obeStatusForDriver = ø,
			accountUpdate = accountU,
			responseAuthenticator = ø
			}
			where accountU = setAccount {
			newCreditLimit = ø,
			newDistanceLimit = {
			dist > THR,
			disUnit = UNIT
			}
			newTimeLimit = ø,
			newDurationLimit = ø,
			newEventLimit = ø,
			newAuthenticator = ø }

5	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract = contract1, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus = accountS2, transactionCounter, mileage, listOfCCCAttributes, authenticator} No distance change shall occur before sending this data element		

6			IF (accountS2 equals to 0)
			THEN TP passed
			ELSE TP failed
			ENDIF

7		⇐	**ChargeReportResponse** = { reportRecipientId = any, dataReceived = (ChargeReport.timeOfReport | ChargeReport.mileage | ChargeReport.transactionCounter), versionsResponse = ø, obeStatusForDriver = 0, accountUpdate = ø, responseAuthenticator = ø}

TP/BEF/FE/BV/11	**Verify that DUT sets Account Status to 'LOW' for distance based on-board accounts**
TP Origin	Specific
Reference	ISO/TS 17575-1, Clause 6.6
Initial Condition	Front End received already Context Data applicable for on-board accounts and with Charge Report Configuration requesting accountStatus, transactionCounter and mileage to be present. 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. No authentication is required.

Stimulus and Expected Behaviour

	DUT		Tester
1	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract = contract1, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus = accountS1, transactionCounter, mileage, listOfCCCAttributes, authenticator}	⇒	
2			Verify structure of sent **ChargeReport**, taking presence and absence of optional 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** = {
			reportRecipientId = any,
			dataReceived = (ChargeReport.timeOfReport |
			ChargeReport.mileage |
			ChargeReport.transactionCounter),
			versionsResponse = ø,
			obeStatusForDriver = ø,
			accountUpdate = accountU,
			responseAuthenticator = ø
			}
			where accountU = setAccount {
			newCreditLimit = ø,
			newDistanceLimit = {
			0 < dist < THR,
			disUnit = UNIT
			}
			newTimeLimit = ø,
			newDurationLimit = ø,
			newEventLimit = ø,
			newAuthenticator = ø }

5	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract = contract1, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus = accountS2, transactionCounter, mileage, listOfCCCAttributes, authenticator} No distance change shall occur before sending this data element		

6			IF (accountS2 equals to 1)
			THEN TP passed
			ELSE TP failed
			ENDIF

7		⇐	**ChargeReportResponse** = { reportRecipientId = any, dataReceived = (ChargeReport.timeOfReport | ChargeReport.mileage | ChargeReport.transactionCounter), versionsResponse = ø, obeStatusForDriver = 0, accountUpdate = ø, responseAuthenticator = ø}

TP/BEF/FE/BV/12	**Verify that DUT sets Account Status to 'EMPTY' for distance based on-board accounts**
TP Origin	Specific
Reference	ISO/TS 17575-1, Clause 6.6
Initial Condition	Front End received already Context Data applicable for on-board accounts and with Charge Report Configuration requesting accountStatus, transactionCounter and mileage to be present. 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. No authentication is required.

Stimulus and Expected Behaviour

	DUT		Tester
1	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract = contract1, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus = accountS1, transactionCounter, mileage, listOfCCCAttributes, authenticator}	⇒	
2			Verify structure of sent **ChargeReport**, taking presence and absence of optional 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** = {
			reportRecipientId = any,
			dataReceived = (ChargeReport.timeOfReport |
			ChargeReport.mileage |
			ChargeReport.transactionCounter),
			versionsResponse = ø,
			obeStatusForDriver = ø,
			accountUpdate = accountU,
			responseAuthenticator = ø
			}
			where accountU = setAccount {
			newCreditLimit = ø,
			newDistanceLimit = {
			dist = 0,
			disUnit = UNIT
			}
			newTimeLimit = ø,
			newDurationLimit = ø,
			newEventLimit = ø,
			newAuthenticator = ø }

5	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract = contract1, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus = accountS2, transactionCounter, mileage, listOfCCCAttributes, authenticator} No distance change shall occur before sending this data element		

6			IF (accountS2 equals to 2)
			THEN TP passed
			ELSE TP failed
			ENDIF

7		⇐	**ChargeReportResponse** = { reportRecipientId = any, dataReceived = (ChargeReport.timeOfReport | ChargeReport.mileage | ChargeReport.transactionCounter), versionsResponse = ø, obeStatusForDriver = 0, accountUpdate = ø, responseAuthenticator = ø}

TP/BEF/FE/BV/13	**Verify that DUT sets Account Status to 'NEGATIVE' for distance based on-board accounts**
TP Origin	Specific
Reference	ISO/TS 17575-1, Clause 6.6
Initial Condition	Front End received already Context Data applicable for on-board accounts and with Charge Report Configuration requesting accountStatus, transactionCounter and mileage to be present. 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. No authentication is required.

Stimulus and Expected Behaviour

	DUT		Tester
1	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract = contract1, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus = accountS1, transactionCounter, mileage, listOfCCCAttributes, authenticator}	⇒	
2			Verify structure of sent **ChargeReport**, taking presence and absence of optional 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** = {
			reportRecipientId = any,
			dataReceived = (ChargeReport.timeOfReport |
			ChargeReport.mileage |
			ChargeReport.transactionCounter),
			versionsResponse = ø,
			obeStatusForDriver = ø,
			accountUpdate = accountU,
			responseAuthenticator = ø
			}
			where accountU = setAccount {
			newCreditLimit = ø,
			newDistanceLimit = {
			dist < 0,
			disUnit = UNIT
			}
			newTimeLimit = ø,
			newDurationLimit = ø,
			newEventLimit = ø,
			newAuthenticator = ø }

5	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract = contract1, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus = accountS2, transactionCounter, mileage, listOfCCCAttributes, authenticator} No distance change shall occur before sending this data element		

6			IF (accountS2 equals to 3)
			THEN TP passed
			ELSE TP failed
			ENDIF

7		⇐	**ChargeReportResponse** = { reportRecipientId = any, dataReceived = (ChargeReport.timeOfReport | ChargeReport.mileage | ChargeReport.transactionCounter), versionsResponse = ø, obeStatusForDriver = 0, accountUpdate = ø, responseAuthenticator = ø}

TP/BEF/FE/BV/14	**Verify that DUT sets Account Status to 'OK' for time based on-board accounts**
TP Origin	Specific
Reference	ISO/TS 17575-1, Clause 6.6
Initial Condition	Front End received already Context Data applicable for on-board accounts and with Charge Report Configuration requesting accountStatus, transactionCounter and mileage to be present. 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. No authentication is required.

Stimulus and Expected Behaviour

	DUT		Tester
1	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract = contract1, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus = accountS1, transactionCounter, mileage, listOfCCCAttributes, authenticator}	⇒	
2			Verify structure of sent **ChargeReport**, taking presence and absence of optional 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** = {
			reportRecipientId = any,
			dataReceived = (ChargeReport.timeOfReport |
			ChargeReport.mileage |
			ChargeReport.transactionCounter),
			versionsResponse = ø,
			obeStatusForDriver = ø,
			accountUpdate = accountU,
			responseAuthenticator = ø
			}
			where accountU = setAccount {
			newCreditLimit = ø,
			newDistanceLimit = ø,
			newTimeLimit > currentTime + THR,
			newDurationLimit = ø,
			newEventLimit = ø,
			newAuthenticator = ø }
			currentTime equals to the timestamp when this step is executed

5	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract = contract1, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus = accountS2, transactionCounter, mileage, listOfCCCAttributes, authenticator} This data unit shall be sent not later than DUT changes its accountStatus.		

6			IF (accountS2 equals to 0)
			THEN TP passed
			ELSE TP failed
			ENDIF

7		⇐	**ChargeReportResponse** = { reportRecipientId = any, dataReceived = (ChargeReport.timeOfReport | ChargeReport.mileage | ChargeReport.transactionCounter), versionsResponse = ø, obeStatusForDriver = 0, accountUpdate = ø, responseAuthenticator = ø}

TP/BEF/FE/BV/15	**Verify that DUT sets Account Status to 'LOW' for time based on-board accounts**
TP Origin	Specific
Reference	ISO/TS 17575-1, Clause 6.6
Initial Condition	Front End received already Context Data applicable for on-board accounts and with Charge Report Configuration requesting accountStatus, transactionCounter and mileage to be present. 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. No authentication is required.

Stimulus and Expected Behaviour

	DUT		Tester
1	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract = contract1, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus = accountS1, transactionCounter, mileage, listOfCCCAttributes, authenticator}	⇒	
2			Verify structure of sent **ChargeReport**, taking presence and absence of optional 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** = {
			reportRecipientId = any,
			dataReceived = (ChargeReport.timeOfReport |
			ChargeReport.mileage |
			ChargeReport.transactionCounter),
			versionsResponse = ø,
			obeStatusForDriver = ø,
			accountUpdate = accountU,
			responseAuthenticator = ø
			}
			where accountU = setAccount {
			newCreditLimit = ø,
			newDistanceLimit = ø,
			currentTime < newTimeLimit < currentTime + THR,
			newDurationLimit = ø,
			newEventLimit = ø,
			newAuthenticator = ø }
			currentTime equals to the timestamp when this step is executed

5	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract = contract1, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus = accountS2, transactionCounter, mileage, listOfCCCAttributes, authenticator} This data unit shall be sent not later than DUT changes its accountStatus.		

6			IF (accountS2 equals to 1)
			THEN TP passed
			ELSE TP failed
			ENDIF

7		⇐	**ChargeReportResponse** = { reportRecipientId = any, dataReceived = (ChargeReport.timeOfReport | ChargeReport.mileage | ChargeReport.transactionCounter), versionsResponse = ø, obeStatusForDriver = 0, accountUpdate = ø, responseAuthenticator = ø}

TP/BEF/FE/BV/16	**Verify that DUT sets Account Status to 'EMPTY' for time based on-board accounts**
TP Origin	Specific
Reference	ISO/TS 17575-1, Clause 6.6
Initial Condition	Front End received already Context Data applicable for on-board accounts and with Charge Report Configuration requesting accountStatus, transactionCounter and mileage to be present. 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. No authentication is required.

Stimulus and Expected Behaviour

	DUT		Tester
1	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract = contract1, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus = accountS1, transactionCounter, mileage, listOfCCCAttributes, authenticator}	⇒	
2			Verify structure of sent **ChargeReport**, taking presence and absence of optional 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** = {
			reportRecipientId = any,
			dataReceived = (ChargeReport.timeOfReport |
			ChargeReport.mileage |
			ChargeReport.transactionCounter),
			versionsResponse = ø,
			obeStatusForDriver = ø,
			accountUpdate = accountU,
			responseAuthenticator = ø
			}
			where accountU = setAccount {
			newCreditLimit = ø,
			newDistanceLimit = ø,
			newTimeLimit = currentTime,
			newDurationLimit = ø,
			newEventLimit = ø,
			newAuthenticator = ø }
			currentTime equals to the timestamp when this step is executed

5	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract = contract1, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus = accountS2, transactionCounter, mileage, listOfCCCAttributes, authenticator} This data unit shall be sent immediately after receipt of ChargeReportResponse. Otherwise Empty status cannot be observed.		

6			IF (accountS2 equals to 2)
			THEN TP passed
			ELSE TP failed
			ENDIF

7		⇐	**ChargeReportResponse** = { reportRecipientId = any, dataReceived = (ChargeReport.timeOfReport | ChargeReport.mileage | ChargeReport.transactionCounter), versionsResponse = ø, obeStatusForDriver = 0, accountUpdate = ø, responseAuthenticator = ø}

TP/BEF/FE/BV/17	**Verify that DUT sets Account Status to 'NEGATIVE' for time based on-board accounts**
TP Origin	Specific
Reference	ISO/TS 17575-1, Clause 6.6
Initial Condition	Front End received already Context Data applicable for on-board accounts and with Charge Report Configuration requesting accountStatus, transactionCounter and mileage to be present. 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. No authentication is required.

Stimulus and Expected Behaviour

	DUT		Tester
1	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract = contract1, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus = accountS1, transactionCounter, mileage, listOfCCCAttributes, authenticator}	⇒	
2			Verify structure of sent **ChargeReport**, taking presence and absence of optional 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** = {
			reportRecipientId = any,
			dataReceived = (ChargeReport.timeOfReport |
			ChargeReport.mileage |
			ChargeReport.transactionCounter),
			versionsResponse = ø,
			obeStatusForDriver = ø,
			accountUpdate = accountU,
			responseAuthenticator = ø
			}
			where accountU = setAccount {
			newCreditLimit = ø,
			newDistanceLimit = ø,
			newTimeLimit < currentTime,
			newDurationLimit = ø,
			newEventLimit = ø,
			newAuthenticator = ø }
			currentTime equals to the timestamp when this step is executed

5	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract = contract1, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus = accountS2, transactionCounter, mileage, listOfCCCAttributes, authenticator}		

6			IF (accountS2 equals to 3)
			THEN TP passed
			ELSE TP failed
			ENDIF

7		⇐	**ChargeReportResponse** = { reportRecipientId = any, dataReceived = (ChargeReport.timeOfReport | ChargeReport.mileage | ChargeReport.transactionCounter), versionsResponse = ø, obeStatusForDriver = 0, accountUpdate = ø, responseAuthenticator = ø}

TP/BEF/FE/BV/18	**Verify that DUT sets Account Status to 'OK' for duration based on-board accounts**
TP Origin	Specific
Reference	ISO/TS 17575-1, Clause 6.6
Initial Condition	Front End received already Context Data applicable for on-board accounts and with Charge Report Configuration requesting accountStatus, transactionCounter and mileage to be present. 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. No authentication is required.

Stimulus and Expected Behaviour

	DUT		Tester
1	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract = contract1, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus = accountS1, transactionCounter, mileage, listOfCCCAttributes, authenticator}	⇒	
2			Verify structure of sent **ChargeReport**, taking presence and absence of optional 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** = {
			reportRecipientId = any,
			dataReceived = (ChargeReport.timeOfReport |
			ChargeReport.mileage |
			ChargeReport.transactionCounter),
			versionsResponse = ø,
			obeStatusForDriver = ø,
			accountUpdate = accountU,
			responseAuthenticator = ø
			}
			where accountU = setAccount {
			newCreditLimit = ø,
			newDistanceLimit = ø,
			newTimeLimit = ø,
			newDurationLimit = {
			dur > THR,
			durUnit = UNIT
			},
			newEventLimit = ø,
			newAuthenticator = ø }

5	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract = contract1, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus = accountS2, transactionCounter, mileage, listOfCCCAttributes, authenticator} This data unit shall be sent not later than DUT changes its accountStatus.		

6			IF (accountS2 equals to 0)
			THEN TP passed
			ELSE TP failed
			ENDIF

7		⇐	**ChargeReportResponse** = { reportRecipientId = any, dataReceived = (ChargeReport.timeOfReport | ChargeReport.mileage | ChargeReport.transactionCounter), versionsResponse = ø, obeStatusForDriver = 0, accountUpdate = ø, responseAuthenticator = ø}

TP/BEF/FE/BV/19	**Verify that DUT sets Account Status to 'LOW' for duration based on-board accounts**
TP Origin	Specific
Reference	ISO/TS 17575-1, Clause 6.6
Initial Condition	Front End received already Context Data applicable for on-board accounts and with Charge Report Configuration requesting accountStatus, transactionCounter and mileage to be present. 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. No authentication is required.

Stimulus and Expected Behaviour

	DUT		Tester
1	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract = contract1, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus = accountS1, transactionCounter, mileage, listOfCCCAttributes, authenticator}	⇒	
2			Verify structure of sent **ChargeReport**, taking presence and absence of optional 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** = {
			reportRecipientId = any,
			dataReceived = (ChargeReport.timeOfReport |
			ChargeReport.mileage |
			ChargeReport.transactionCounter),
			versionsResponse = ø,
			obeStatusForDriver = ø,
			accountUpdate = accountU,
			responseAuthenticator = ø
			}
			where accountU = setAccount {
			newCreditLimit = ø,
			newDistanceLimit = ø,
			newTimeLimit = ø,
			newDurationLimit = {
			0 < dur < THR,
			durUnit = UNIT
			},
			newEventLimit = ø,
			newAuthenticator = ø }

5	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract = contract1, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus = accountS2, transactionCounter, mileage, listOfCCCAttributes, authenticator} This data unit shall be sent not later than DUT changes its accountStatus.		

6			IF (accountS2 equals to 1)
			THEN TP passed
			ELSE TP failed
			ENDIF

7		⇐	**ChargeReportResponse** = { reportRecipientId = any, dataReceived = (ChargeReport.timeOfReport | ChargeReport.mileage | ChargeReport.transactionCounter), versionsResponse = ø, obeStatusForDriver = 0, accountUpdate = ø, responseAuthenticator = ø}

TP/BEF/FE/BV/20	**Verify that DUT sets Account Status to 'EMPTY' for duration based on-board accounts**
TP Origin	Specific
Reference	ISO/TS 17575-1, Clause 6.6
Initial Condition	Front End received already Context Data applicable for on-board accounts and with Charge Report Configuration requesting accountStatus, transactionCounter and mileage to be present. 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. No authentication is required.

Stimulus and Expected Behaviour

	DUT		Tester
1	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract = contract1, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus = accountS1, transactionCounter, mileage, listOfCCCAttributes, authenticator}	⇒	
2			Verify structure of sent **ChargeReport**, taking presence and absence of optional 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** = {
			reportRecipientId = any,
			dataReceived = (ChargeReport.timeOfReport |
			ChargeReport.mileage |
			ChargeReport.transactionCounter),
			versionsResponse = ø,
			obeStatusForDriver = ø,
			accountUpdate = accountU,
			responseAuthenticator = ø
			}
			where accountU = setAccount {
			newCreditLimit = ø,
			newDistanceLimit = ø,
			newTimeLimit = ø,
			newDurationLimit = {
			dur = 0,
			durUnit = UNIT
			},
			newEventLimit = ø,
			newAuthenticator = ø }

5	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract = contract1, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus = accountS2, transactionCounter, mileage, listOfCCCAttributes, authenticator} This data unit shall be sent not later than DUT changes its accountStatus.		

6			IF (accountS2 equals to 2)
			THEN TP passed
			ELSE TP failed
			ENDIF

7		⇐	**ChargeReportResponse** = { reportRecipientId = any, dataReceived = (ChargeReport.timeOfReport | ChargeReport.mileage | ChargeReport.transactionCounter), versionsResponse = ø, obeStatusForDriver = 0, accountUpdate = ø, responseAuthenticator = ø}

TP/BEF/FE/BV/21	**Verify that DUT sets Account Status to 'NEGATIVE' for duration based on-board accounts**
TP Origin	Specific
Reference	ISO/TS 17575-1, Clause 6.6
Initial Condition	Front End received already Context Data applicable for on-board accounts and with Charge Report Configuration requesting accountStatus, transactionCounter and mileage to be present. 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. No authentication is required.

Stimulus and Expected Behaviour

	DUT		Tester
1	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract = contract1, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus = accountS1, transactionCounter, mileage, listOfCCCAttributes, authenticator}	⇒	
2			Verify structure of sent **ChargeReport**, taking presence and absence of optional 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** = {
			reportRecipientId = any,
			dataReceived = (ChargeReport.timeOfReport |
			ChargeReport.mileage |
			ChargeReport.transactionCounter),
			versionsResponse = ø,
			obeStatusForDriver = ø,
			accountUpdate = accountU,
			responseAuthenticator = ø
			}
			where accountU = setAccount {
			newCreditLimit = ø,
			newDistanceLimit = ø,
			newTimeLimit = ø,
			newDurationLimit = {
			dur < 0,
			durUnit = UNIT
			},
			newEventLimit = ø,
			newAuthenticator = ø }

5	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract = contract1, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus = accountS2, transactionCounter, mileage, listOfCCCAttributes, authenticator} This data unit shall be sent not later than DUT changes its accountStatus.		

6			IF (accountS2 equals to 3)
			THEN TP passed
			ELSE TP failed
			ENDIF

7		⇐	**ChargeReportResponse** = { reportRecipientId = any, dataReceived = (ChargeReport.timeOfReport | ChargeReport.mileage | ChargeReport.transactionCounter), versionsResponse = ø, obeStatusForDriver = 0, accountUpdate = ø, responseAuthenticator = ø}

TP/BEF/FE/BV/22	**Verify that DUT sets Account Status to 'OK' for event based on-board accounts**
TP Origin	Specific
Reference	ISO/TS 17575-1, Clause 6.6
Initial Condition	Front End received already Context Data applicable for on-board accounts and with Charge Report Configuration requesting accountStatus, transactionCounter and mileage to be present. Front End implementation supports on-board account. Threshold for on-board account is set to THR. Front End is initialized and has a toll context activated. No authentication is required.

Stimulus and Expected Behaviour

	DUT		Tester
1	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract = contract1, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus = accountS1, transactionCounter, mileage, listOfCCCAttributes, authenticator}	⇒	
2			Verify structure of sent **ChargeReport**, taking presence and absence of optional 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** = {
			reportRecipientId = any,
			dataReceived = (ChargeReport.timeOfReport |
			ChargeReport.mileage |
			ChargeReport.transactionCounter),
			versionsResponse = ø,
			obeStatusForDriver = ø,
			accountUpdate = accountU,
			responseAuthenticator = ø
			}
			
			where accountU = setAccount {
			newCreditLimit = ø,
			newDistanceLimit = ø,
			newTimeLimit = ø,
			newDurationLimit = ø,
			newEventLimit > THR,
			newAuthenticator = ø }

5	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract = contract1, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus = accountS2, transactionCounter, mileage, listOfCCCAttributes, authenticator} No accountable events shall occur before sending this data element		

6			IF (accountS2 equals to 0)
			THEN TP passed
			ELSE TP failed
			ENDIF

7		⇐	**ChargeReportResponse** = { reportRecipientId = any, dataReceived = (ChargeReport.timeOfReport | ChargeReport.mileage | ChargeReport.transactionCounter), versionsResponse = ø, obeStatusForDriver = 0, accountUpdate = ø, responseAuthenticator = ø}

TP/BEF/FE/BV/23	**Verify that DUT sets Account Status to 'LOW' for event based on-board accounts**
TP Origin	Specific
Reference	ISO/TS 17575-1, Clause 6.6
Initial Condition	Front End received already Context Data applicable for on-board accounts and with Charge Report Configuration requesting accountStatus, transactionCounter and mileage to be present. Front End implementation supports on-board account. Threshold for on-board account is set to THR. Front End is initialized and has a toll context activated. No authentication is required.

Stimulus and Expected Behaviour

	DUT		Tester
1	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract = contract1, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus = accountS1, transactionCounter, mileage, listOfCCCAttributes, authenticator}	⇒	
2			Verify structure of sent **ChargeReport**, taking presence and absence of optional 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** = {
			reportRecipientId = any,
			dataReceived = (ChargeReport.timeOfReport |
			ChargeReport.mileage |
			ChargeReport.transactionCounter),
			versionsResponse = ø,
			obeStatusForDriver = ø,
			accountUpdate = accountU,
			responseAuthenticator = ø
			}
			
			where accountU = setAccount {
			newCreditLimit = ø,
			newDistanceLimit = ø,
			newTimeLimit = ø,
			newDurationLimit = ø,
			0 < newEventLimit < THR,
			newAuthenticator = ø }

5	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract = contract1, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus = accountS2, transactionCounter, mileage, listOfCCCAttributes, authenticator} No accountable events shall occur before sending this data element		

6			IF (accountS2 equals to 1)
			THEN TP passed
			ELSE TP failed
			ENDIF

7		⇐	**ChargeReportResponse** = { reportRecipientId = any, dataReceived = (ChargeReport.timeOfReport | ChargeReport.mileage | ChargeReport.transactionCounter), versionsResponse = ø, obeStatusForDriver = 0, accountUpdate = ø, responseAuthenticator = ø}

TP/BEF/FE/BV/24	**Verify that DUT sets Account Status to 'EMPTY' for event based on-board accounts**
TP Origin	Specific
Reference	ISO/TS 17575-1, Clause 6.6
Initial Condition	Front End received already Context Data applicable for on-board accounts and with Charge Report Configuration requesting accountStatus, transactionCounter and mileage to be present. Front End implementation supports on-board account. Threshold for on-board account is set to THR. Front End is initialized and has a toll context activated. No authentication is required.

Stimulus and Expected Behaviour

	DUT		Tester
1	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract = contract1, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus = accountS1, transactionCounter, mileage, listOfCCCAttributes, authenticator}	⇒	
2			Verify structure of sent **ChargeReport**, taking presence and absence of optional 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** = {
			reportRecipientId = any,
			dataReceived = (ChargeReport.timeOfReport |
			ChargeReport.mileage |
			ChargeReport.transactionCounter),
			versionsResponse = ø,
			obeStatusForDriver = ø,
			accountUpdate = accountU,
			responseAuthenticator = ø
			}
			
			where accountU = setAccount {
			newCreditLimit = ø,
			newDistanceLimit = ø,
			newTimeLimit = ø,
			newDurationLimit = ø,
			newEventLimit = 0,
			newAuthenticator = ø }

5	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract = contract1, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus = accountS2, transactionCounter, mileage, listOfCCCAttributes, authenticator} No accountable events shall occur before sending this data element		

6			IF (accountS2 equals to 2)
			THEN TP passed
			ELSE TP failed
			ENDIF

7		⇐	**ChargeReportResponse** = { reportRecipientId = any, dataReceived = (ChargeReport.timeOfReport | ChargeReport.mileage | ChargeReport.transactionCounter), versionsResponse = ø, obeStatusForDriver = 0, accountUpdate = ø, responseAuthenticator = ø}

TP/BEF/FE/BV/25	**Verify that DUT sets Account Status to 'NEGATIVE' for event based on-board accounts**
TP Origin	Specific
Reference	ISO/TS 17575-1, Clause 6.6
Initial Condition	Front End received already Context Data applicable for on-board accounts and with Charge Report Configuration requesting accountStatus, transactionCounter and mileage to be present. Front End implementation supports on-board account. Threshold for on-board account is set to THR. Front End is initialized and has a toll context activated. No authentication is required.

Stimulus and Expected Behaviour

	DUT		Tester
1	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract = contract1, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus = accountS1, transactionCounter, mileage, listOfCCCAttributes, authenticator}	⇒	
2			Verify structure of sent **ChargeReport**, taking presence and absence of optional 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** = {
			reportRecipientId = any,
			dataReceived = (ChargeReport.timeOfReport |
			ChargeReport.mileage |
			ChargeReport.transactionCounter),
			versionsResponse = ø,
			obeStatusForDriver = ø,
			accountUpdate = accountU,
			responseAuthenticator = ø
			}
			
			where accountU = setAccount {
			newCreditLimit = ø,
			newDistanceLimit = ø,
			newTimeLimit = ø,
			newDurationLimit = ø,
			newEventLimit < 0,
			newAuthenticator = ø }

5	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract = contract1, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus = accountS2, transactionCounter, mileage, listOfCCCAttributes, authenticator} No accountable events shall occur before sending this data element		

6			IF (accountS2 equals to 3)
			THEN TP passed
			ELSE TP failed
			ENDIF

7		⇐	**ChargeReportResponse** = { reportRecipientId = any, dataReceived = (ChargeReport.timeOfReport | ChargeReport.mileage | ChargeReport.transactionCounter), versionsResponse = ø, obeStatusForDriver = 0, accountUpdate = ø, responseAuthenticator = ø}

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/TS 17575-1, Clause 6.6
Initial Condition	Front End received already Context Data applicable for on-board accounts and with Charge Report Configuration requesting accountStatus, transactionCounter and mileage to be present. 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. Front End shall not utilize any credits during the whole test purpose. No authentication is required.

Stimulus and Expected Behaviour

	DUT		Tester
1	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract = contract1, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus = accountS1, transactionCounter, mileage, listOfCCCAttributes, authenticator}	⇒	
2			Verify structure of sent **ChargeReport**, taking presence and absence of optional 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** = {
			reportRecipientId = any,
			dataReceived = (ChargeReport.timeOfReport |
			ChargeReport.mileage |
			ChargeReport.transactionCounter),
			versionsResponse = ø,
			obeStatusForDriver = ø,
			accountUpdate = accountU,
			responseAuthenticator = ø
			}
			
			where accountU = addToAccount {
			addCredit = {
			0< paymentFeeAmount < THR,
			paymentFeeUnit = UNIT
			}
			addDistance = ø,
			addTime = ø,
			addDuration = ø,
			addEvents = ø,
			addAuthenticator = ø }

5	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract = contract1, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus = accountS2, transactionCounter, mileage, listOfCCCAttributes, authenticator}		
6			Verify structure of sent **ChargeReport**, taking presence and absence of optional 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 → 3
			3 → 2
			3 → 1
			2 → 1
			1 → 1
			1 → 0
			0 → 0
			
			IF (transition not allowed)
			THEN TP failed
			ELSE GOTO STEP 9
			ENDIF

9			IF (accountS2 equals to 0)
			THEN TP passed
			ELSE {
			accountS1=accountS2
			GOTO STEP 4
			ENDIF

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/TS 17575-1, Clause 6.6
Initial Condition	Front End received already Context Data applicable for on-board accounts and with Charge Report Configuration requesting accountStatus, transactionCounter and mileage to be present. 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. Front End shall not change its mileage during the whole test purpose. No authentication is required.

Stimulus and Expected Behaviour

	DUT		Tester
1	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract = contract1, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus = accountS1, transactionCounter, mileage, listOfCCCAttributes, authenticator}	⇒	
2			Verify structure of sent **ChargeReport**, taking presence and absence of optional 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** = {
			reportRecipientId = any,
			dataReceived = (ChargeReport.timeOfReport |
			ChargeReport.mileage |
			ChargeReport.transactionCounter),
			versionsResponse = ø,
			obeStatusForDriver = ø,
			accountUpdate = accountU,
			responseAuthenticator = ø
			}
			
			where accountU = addToAccount {
			addCredit = ø,
			addDistance = {
			0 < dist < THR,
			disUnit = UNIT
			},
			addTime = ø,
			addDuration = ø,
			addEvents = ø,
			addAuthenticator = ø }

5	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract = contract1, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus = accountS2, transactionCounter, mileage, listOfCCCAttributes, authenticator}		
6			Verify structure of sent **ChargeReport**, taking presence and absence of optional 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 → 3
			3 → 2
			3 → 1
			2 → 1
			1 → 1
			1 → 0
			0 → 0
			
			IF (transition not allowed)
			THEN TP failed
			ELSE GOTO STEP 9
			ENDIF

9			IF (accountS2 equals to 0)
			THEN TP passed
			ELSE {
			accountS1=accountS2
			GOTO STEP 4
			ENDIF

TP/BEF/FE/BV/28	**Verify that DUT updates Account Status for event based on-board accounts upon AddToAccount operation**
TP Origin	Specific
Reference	ISO/TS 17575-1, Clause 6.6
Initial Condition	Front End received already Context Data applicable for on-board accounts and with Charge Report Configuration requesting accountStatus, transactionCounter and mileage to be present. Front End implementation supports on-board account. On-board account is event 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. No accountable event happens during whole test purpose. No authentication is required.

Stimulus and Expected Behaviour

	DUT		Tester
1	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract = contract1, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus = accountS1, transactionCounter, mileage, listOfCCCAttributes, authenticator}	⇒	
2			Verify structure of sent **ChargeReport**, taking presence and absence of optional 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** = {
			reportRecipientId = any,
			dataReceived = (ChargeReport.timeOfReport |
			ChargeReport.mileage |
			ChargeReport.transactionCounter),
			versionsResponse = ø,
			obeStatusForDriver = ø,
			accountUpdate = accountU,
			responseAuthenticator = ø
			}
			
			where accountU = addToAccount {
			addCredit = ø,
			addDistance= ø,
			addTime = ø,
			addDuration = ø,
			0 < addEvents < THR,
			addAuthenticator = ø }

5	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract = contract1, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus = accountS2, transactionCounter, mileage, listOfCCCAttributes, authenticator}		
6			Verify structure of sent **ChargeReport**, taking presence and absence of optional 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 → 3
			3 → 2
			3 → 1
			2 → 1
			1 → 1
			1 → 0
			0 → 0
			
			IF (transition not allowed)
			THEN TP failed
			ELSE GOTO STEP 9
			ENDIF

9			IF (accountS2 equals to 0)
			THEN TP passed
			ELSE {
			accountS1=accountS2
			GOTO STEP 4
			ENDIF

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/TS 17575-1, Clause 6.6.2
Initial Condition	Front End received already Context Data applicable for on-board accounts and with Charge Report Configuration requesting accountStatus, transactionCounter and mileage to be present. Front End implementation supports credit based on-board account. 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. Front End shall not utilize any credits during the whole test purpose. No authentication is required.

Stimulus and Expected Behaviour

	DUT		Tester
1	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract = contract1, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus = accountS1, transactionCounter, mileage, listOfCCCAttributes, authenticator}	⇒	
2			Verify structure of sent **ChargeReport**, taking presence and absence of optional 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** = {
			reportRecipientId = any,
			dataReceived = (ChargeReport.timeOfReport |
			ChargeReport.mileage |
			ChargeReport.transactionCounter),
			versionsResponse = ø,
			obeStatusForDriver = ø,
			accountUpdate = accountU,
			responseAuthenticator = ø
			}
			
			where accountU = reloadAccount {
			reloadOldCreditAmount = T,
			reloadOldDistanceLimit = ø,
			reloadOldTimeLimit = ø,
			reloadOldDurationLimit = ø,
			reloadOldEventLimit = ø,
			reloadAuthenticator = ø }

5	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract = contract1, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus = accountS2, transactionCounter, mileage, listOfCCCAttributes, authenticator}		
6			Verify structure of sent **ChargeReport**, taking presence and absence of optional 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 → 3
			3 → 2
			3 → 1
			2 → 1
			1 → 1
			1 → 0
			0 → 0
			
			IF (transition not allowed)
			THEN TP failed
			ELSE GOTO STEP 9
			ENDIF

9			IF (accountS2 equals to 0)
			THEN TP passed
			ELSE {
			accountS1=accountS2
			GOTO STEP 4
			ENDIF

TP/BEF/FE/BV/30	**Verify that DUT updates Account Status for distance based on-board accounts upon ReloadAccount operation**
TP Origin	Specific
Reference	ISO/TS 17575-1, Clause 6.6.2
Initial Condition	Front End received already Context Data applicable for on-board accounts and with Charge Report Configuration requesting accountStatus, transactionCounter and mileage to be present. Front End implementation supports distance based on-board account. 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. Front End shall not change its mileage during the whole test purpose. No authentication is required.

Stimulus and Expected Behaviour

	DUT		Tester
1	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract = contract1, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus = accountS1, transactionCounter, mileage, listOfCCCAttributes, authenticator}	⇒	
2			Verify structure of sent **ChargeReport**, taking presence and absence of optional 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** = {
			reportRecipientId = any,
			dataReceived = (ChargeReport.timeOfReport |
			ChargeReport.mileage |
			ChargeReport.transactionCounter),
			versionsResponse = ø,
			obeStatusForDriver = ø,
			accountUpdate = accountU,
			responseAuthenticator = ø
			}
			where accountU = reloadAccount {
			reloadOldCreditAmount = ø,
			reloadOldDistanceLimit = T,
			reloadOldTimeLimit = ø,
			reloadOldDurationLimit = ø,
			reloadOldEventLimit = ø,
			reloadAuthenticator = ø }

5	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract = contract1, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus = accountS2, transactionCounter, mileage, listOfCCCAttributes, authenticator}		
6			Verify structure of sent **ChargeReport**, taking presence and absence of optional 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 → 3
			3 → 2
			3 → 1
			2 → 1
			1 → 1
			1 → 0
			0 → 0
			
			IF (transition not allowed)
			THEN TP failed
			ELSE GOTO STEP 9
			ENDIF

9			IF (accountS2 equals to 0)
			THEN TP passed
			ELSE {
			accountS1=accountS2
			GOTO STEP 4
			ENDIF

TP/BEF/FE/BV/31	**Verify that DUT updates Account Status for event based on-board accounts upon ReloadAccount operation**
TP Origin	Specific
Reference	ISO/TS 17575-1, Clause 6.6.2
Initial Condition	Front End received already Context Data applicable for on-board accounts and with Charge Report Configuration requesting accountStatus, transactionCounter and mileage to be present. Front End implementation supports event based on-board account. 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. No authentication is required.

Stimulus and Expected Behaviour

	DUT		Tester
1	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract = contract1, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus = accountS1, transactionCounter, mileage, listOfCCCAttributes, authenticator}	⇒	
2			Verify structure of sent **ChargeReport**, taking presence and absence of optional 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** = {
			reportRecipientId = any,
			dataReceived = (ChargeReport.timeOfReport |
			ChargeReport.mileage |
			ChargeReport.transactionCounter),
			versionsResponse = ø,
			obeStatusForDriver = ø,
			accountUpdate = accountU,
			responseAuthenticator = ø
			}
			where accountU = reloadAccount {
			reloadOldCreditAmount = ø,
			reloadOldDistanceLimit = ø,
			reloadOldTimeLimit = ø,
			reloadOldDurationLimit = ø,
			reloadOldEventLimit = T,
			reloadAuthenticator = ø }

5	**ChargeReport** = { obeId, vehicleLPNr, paymentMeans, serviceProviderContract = contract1, tollCharger, timeOfReport, reportPeriod, versionInfo, usageStatementList, vatForThisSession, accountStatus = accountS2, transactionCounter, mileage, listOfCCCAttributes, authenticator}		
6			Verify structure of sent **ChargeReport**, taking presence and absence of optional 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 → 3
			3 → 2
			3 → 1
			2 → 1
			1 → 1
			1 → 0
			0 → 0
			
			IF (transition not allowed)
			THEN TP failed
			ELSE GOTO STEP 9
			ENDIF

9			IF (accountS2 equals to 0)
			THEN TP passed
			ELSE {
			accountS1=accountS2
			GOTO STEP 4
			ENDIF