



BSI Standards Publication

Electronic fee collection — Evaluation of equipment for conformity to ISO/TS 17575-2

Part 1: Test suite structure and test purposes

NO COPYING WITHOUT BSI PERMISSION EXCEPT AS PERMITTED BY COPYRIGHT LAW

raising standards worldwide™

Full document details: PD CEN ISO/TS 16401-1:2012

National foreword

This Published Document is the UK implementation of CEN ISO/TS 16401-1:2012.

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

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

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

© The British Standards Institution 2012

Published by BSI Standards Limited 2012

ISBN 978 0 580 71831 1

ICS 03.220.20; 35.240.60

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

This Published Document was published under the authority of the Standards Policy and Strategy Committee on 30 April 2012.

Amendments issued since publication

Amd. No.	Date	Text affected
----------	------	---------------

TECHNICAL SPECIFICATION
SPÉCIFICATION TECHNIQUE
TECHNISCHE SPEZIFIKATION

CEN ISO/TS 16401-1

March 2012

ICS 03.220.20; 35.240.60

English Version

**Electronic fee collection - Evaluation of equipment for conformity
to ISO/TS 17575-2 - Part 1: Test suite structure and test
purposes (ISO 16401-1:2012)**

Perception du télépéage - Évaluation de conformité de
l'équipement à l'ISO/TS 17575-2 - Partie 1: Structure de la
suite d'essais et objectifs d'essai (ISO 16401-1:2012)

Elektronische Gebührenerhebung -
Konformitätsevaluierung von Einrichtungen nach CEN
ISO/TS 17575-2 - Teil 1: Prüfreihe Struktur und
Prüfabichten (ISO 16401-1:2012)

This Technical Specification (CEN/TS) was approved by CEN on 30 January 2012 for provisional application.

The period of validity of this CEN/TS is limited initially to three years. After two years the members of CEN will be requested to submit their comments, particularly on the question whether the CEN/TS can be converted into a European Standard.

CEN members are required to announce the existence of this CEN/TS in the same way as for an EN and to make the CEN/TS available promptly at national level in an appropriate form. It is permissible to keep conflicting national standards in force (in parallel to the CEN/TS) until the final decision about the possible conversion of the CEN/TS into an EN is reached.

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



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

Management Centre: Avenue Marnix 17, B-1000 Brussels

Contents	Page
Foreword.....	3

Foreword

This document (CEN ISO/TS 16401-1:2012) has been prepared by Technical Committee CEN/TC 278 "Road transport and traffic telematics", the secretariat of which is held by NEN, in collaboration with Technical Committee ISO/TC 204 "Intelligent transport systems".

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

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

Contents

	Page
Introduction.....	v
1 Scope.....	1
2 Normative references.....	1
3 Terms and definitions	1
4 Abbreviations.....	2
5 Test Suite Structure (TSS).....	3
5.1 Structure.....	3
5.2 Reference to conformance test specifications	3
5.3 Test Purposes (TP).....	4
5.3.1 TP definition conventions.....	4
5.3.2 TP naming conventions	4
5.4 Protocol Conformance Test Report (PCTR)	5
Annex A (normative) TP for Front End Communications API.....	6
Annex B (normative) Annex ATP for Front End Application.....	139
Annex C (informative) PCTR Proforma for Front End Communications API.....	143
Annex D (informative) PCTR Proforma for Front End Application	150
Bibliography.....	154

Introduction

This part of ISO/TS 16401 is part of a set of standards that supports interoperability of autonomous 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 Communications API and *Front End application* that comply with the requirements in ISO/TS 17575-2.

This part of ISO/TS 16401 is intended to

- assess Front End Communications API and *Front End application* capabilities,
- assess Front End Communications API and *Front End application* behaviour,
- serve as a guide for Front End Communications API and *Front End application* conformance evaluation and type approval,
- achieve comparability between the results of the corresponding tests applied in different places at different times, and
- facilitate communications between parties.

This part of ISO/TS 16401 is based on

- ISO/TS 17575-2, and
- the ISO/IEC 9646 family of standards on conformance test methodology.

Electronic fee collection — Evaluation of equipment for conformity to ISO/TS 17575-2 —

Part 1: Test suite structure and test purposes

1 Scope

This part of ISO/TS 16401 specifies the test suite structure (TSS) and test purposes (TP) to evaluate the conformity of Front End Communications API and *Front End application* to ISO/TS 17575-2.

The objective of this part of ISO/TS 16401 is to provide a basis for conformance tests for Front End Communications API and *Front End application* in Electronic Fee Collection based on autonomous on-board equipment (OBE) to enable interoperability between different equipment supplied by different manufacturers.

This part of ISO/TS 16401 covers the test purposes for Front End Communications API covering functionalities related to instance handling, session handling, communication service primitives (i.e. sending/receiving of ADUs) and visible state transitions. It fully covers EFC communication services claimed in ISO/TS 17575-2 clause 7 and PICS proforma Clause B.2 ISO/TS 17575-2. Claims related to Front End Storage capacity are outside of the scope of this part of ISO/TS 16401.

This part of ISO/TS 16401 covers the test purposes for *Front End application* related to session establishment on *Back End* request and related to session re-establishment when session requested by *Back End* failed. There are no other claims with respect to *Front End application* claimed in ISO/TS 17575-2.

The underlying communication technology requirements for layer 1-4 specified in Clause 8 ISO/TS 17575-2 is outside of the scope of this part of ISO/TS 16401.

Similarly *Back End* communications API is outside of the scope of this part of ISO/TS 16401. According to ISO/TS 17575-2 it is expected that these Front End Communications API will be reflected in the BE, however BE Communications API is outside of the scope of ISO/TS 17575-2.

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/TS 17575-1, *Electronic fee collection — Application interface definition for autonomous systems — Part 2: Charging*

ISO/TS 17575-2, *Electronic fee collection — Application interface definition for autonomous systems — Part 2: Communication and connection to the lower layers*

3 Terms and definitions

For the purposes of this document, the terms and definitions given in ISO/TS 17575-1 and the following apply.

3.1

contract

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

[ISO 14906:2011, definition 3.7]

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

3.2

Front End application

part of the Front End above the API

3.3

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.

3.4

toll charger

legal entity charging toll for vehicles in a *toll domain*

[ISO/TS 17574:2009, definition 3.27]

4 Abbreviations

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

ADU	Application Data Unit
API	Application Programming Interface
ASN.1	Abstract Syntax Notation One
ATS	Abstract Test Suite
BE	<i>Back End</i>
BI	Invalid Behaviour
BV	Valid Behaviour
CCC	Compliance Check Communication
CN	Cellular Network
DUT	Device Under Tests
EFC	Electronic Fee Collection
FE	Front End
GNSS	Global Navigation Satellite Systems
HMI	Human Machine Interface
ID	Identifier

OBE	On-Board Equipment
PCTR	Protocol Conformance Test Report
PICS	Protocol Implementation Conformance Statements
TP	Test Purposes
TSS	Test Suite Structure
VAT	Value Added Tax

5 Test Suite Structure (TSS)

5.1 Structure

Table 1 — Test Suite Structures shows the Test Suite Structure (TSS).

Table 1 — Test Suite Structures

Group	Type of DUT	Behaviour
Instance Handling	Front End Communications API	Valid Behaviour
		Invalid Behaviour
Session Handling	Front End Communications API	Valid Behaviour
		Invalid Behaviour
	<i>Front End application</i>	Valid Behaviour
Communication Service Primitives	Front End Communications API	Valid Behaviour
		Invalid Behaviour
State Transitions	Front End Communications API	Valid Behaviour

5.2 Reference to conformance test specifications

This part of ISO/TS 16401 takes into account already defined test purposes for conformance to the base standards by referencing them, so that:

- For test purposes that are **identical** to those defined in the base standards conformance test cases direct reference is reported. For reader's convenience, the title or a verbal description of the referenced test purpose is given, together with the reference.
- For test purposes that are **derived** from those defined in the base standards conformance test cases, a direct reference is reported, plus an indication on how the referred test purpose has to be modified for the profile conformance testing.
- For test purposes that are **specific** to ISO/TS 17575-2, a complete description is given.
- 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 — TP Definition Rules below. All Test Purposes are defined in Annex A and Annex B, including the special notation and symbol conventions that shall be used.

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. API or APPL);

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>		
<i>applicable for FE Communications API</i>	IH	Instance Handling
<i>applicable for FE Communications API</i>	SH	Session Handling
<i>applicable for FE Application</i>	SH	Session Handling
<i>applicable for FE Communications API</i>	CSP	Communications Service Primitives
<i>applicable for FE Communications API</i>	ST	State Transitions
<dut> = type of DUT	API	Front End Communications API
	APPL	<i>Front End application</i>
x = Type of testing	BV	Valid Behaviour Tests
	BI	Invalid Behaviour Tests
<nn> = sequential number	(01-99)	Test Purpose Number

5.4 Protocol Conformance Test Report (PCTR)

The supplier of the Front End and Back End, respectively, is responsible for providing a Protocol Conformance Test Report (PCTR).

The supplier of the Front End and the Back End shall complete a PCTR; see Annex C and Annex D for the proformas.

Annex A (normative)

TP for Front End Communications API

A.1 Introduction

This annex contains the Test Purposes (TP) for the conformity evaluation of Front End Communications API to ISO/TS 17575-2.

A.1.1 TP symbols conventions

A special notation and symbol convention is 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(Type1=value1) □	The Tester executes an XXX method of Front End Communications API with argument of Type1 having value value1. Value1 shall be stored in tester's memory for further TP processing.
□ R:ReturnedType	The DUT returns a value of type ReturnedType
□ C:CallbackName (Type1)	The DUT provides a callback CallbackName receiving variable of type Type1.
Type ISO/TS 17575-2	Anytime Type defined in ISO/TS 17575-2 is used, it means a variable of Type.
$A \rightarrow B$	A “is transformed” into B
∅	Means “empty” or “not set”.
$A \mid B$	A OR B
$A \neq B$	A is not equal B
$i = i+1$	Increment variable i by 1

Testing Front End Communications API it is needed to trigger operations and observe the DUT feedback both from the *Front End application* and Remote End (i.e. *Back End*) perspective. Thus there are two test points located as shown in Figure A.1.

Application emulation test point is using directly with DUT and emulates the *Front End application* layer. It is identified in the following test purposes by AppEm discriminator.

Remote End emulation test point is linked with DUT over communications channel. Depending on the test purposes it emulates application, presentation and session layer. It is identified in the following test purposes by RemEnd discriminator.

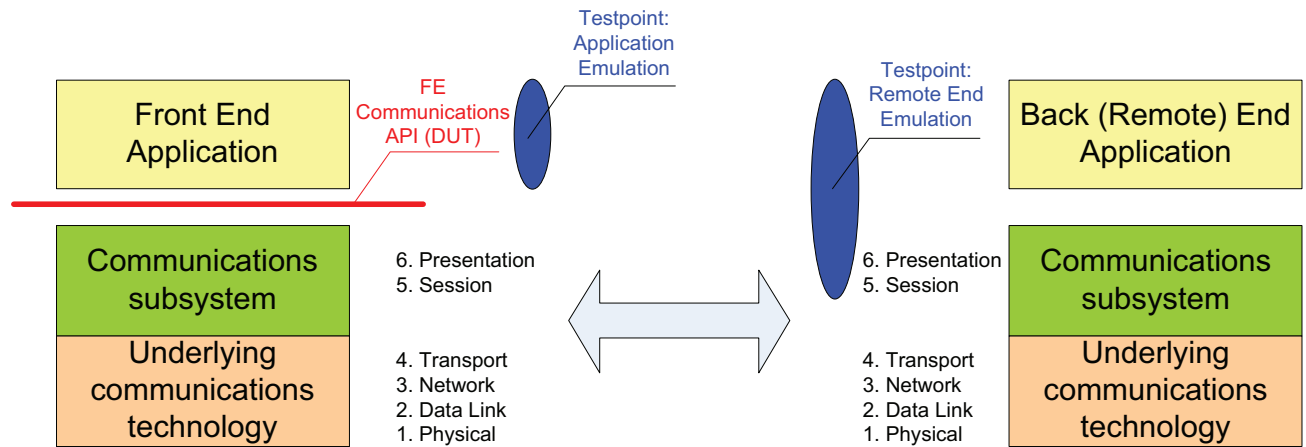


Figure A.1 — Handling of ADUs applicable for particular TP

A.2 Instance Handling

These Test Purposes apply to instance handling as claimed in ISO/TS 17575-2 clause B.2 with respect to following PICS proforma entries:

- API supports InitialiseInstance;
- API supports SetParameter;
- API supports GetParameter;
- API supports DeleteParameter;
- API supports DropInstance;
- API supports StackAvail.

A.2.1 BV test purposes (Valid Behaviour tests)

Test subgroup objective:

- to test DUT behaviour with respect to instance initialization including multiple instance handling in parallel;
- to test DUT behaviour with respect to parameter setting and updating;
- to test DUT behaviour with respect to parameter getting;
- to test DUT behaviour with respect to parameter deleting;
- to test DUT behaviour with respect to availability of communications stack;

- to test DUT behaviour with respect to dropping the session with following severities:
- SENormal;
- SEUrgent;
- SEUnconditional.

TP/IH/API/BV/01	Verify the communications interface initialization			
TP Origin	Specific			
Reference	ISO/TS 17575-2, Clause 7.2.1			
Initial Condition	Front End Communications API must handle at least one underlying communications stack which StackID equals to stack1. Set of Callback instances is instantiated.			
Stimulus and Expected Behaviour				
	Tester	Test Point		DUT
1	InitialiseInstance (StackID = stack1, Callbacks = cb1)	AppEm	☐	
2		AppEm	☐	R: Instace
3	Verify whether Instance is valid			
4	IF verify OK THEN TP passed ELSE TP failed ENDI			

TP/IH/API/BV/02	Verify the multiple instance communications interface initialization based on the same communications stack			
TP Origin	Specific			
Reference	ISO/TS 17575-2, Clause 7.2.1			
Initial Condition	Front End Communications API must handle at least one underlying communications stack which StackID equals to stack1. Sets of Callback instances are instantiated.			
Stimulus and Expected Behaviour				
	Tester	Test Point		DUT
1	InitialiseInstance (StackID = stack1, Callbacks = cb1)	AppEm	☐	
2		AppEm	☐	R: Instance
3	Verify whether Instance is valid			
4	IF verify OK THEN GOTO step5 ELSE TP failed ENDIF			
5	InitialiseInstance (StackID = stack1, Callbacks = cb2)	AppEm	☐	
6		AppEm	☐	R: Instance
7	Verify whether Instance is vali			
8	IF verify OK THEN GOTO step 9 ELSE TP failed ENDIF			
9	InitialiseInstance (StackID = stack1, Callbacks = cb3)	AppEm	☐	
10		AppEm	☐	Instance
11	Verify whether Instance ☐ s valid			
12	IF verify OK THEN TP passed ELSE TP failed ENDIF			

TP/IH/API/BV/03	Verify the multiple instance communications interface initialization based on different communications stack			
TP Origin	Specific			
Reference	ISO/TS 17575-2, Clause 7.2.1			
Initial Condition	Front End Communications API must handle at least 2 underlying communications stacks which StackID equals to stack1 and stack2. Sets of Callback instances are instantiated.			
Stimulus and Expected Behaviour				
	Tester	Test Point		DUT
1	InitialiseInstance (StackID = stack1, Callbacks = cb1)	AppEm	☐	
2		AppEm	☐	R: Instance
3	Verify whether Instance is valid			
4	IF verify OK THEN GOTO step5 ELSE TP failed ENDIF			
5	InitialiseInstance (StackID = stack2, Callbacks = cb2)	AppEm	☐	
6		AppEm	☐	R: Instance
7	Verify whether Instance is vali			
8	IF verify OK THEN TP passed ELSE TP failed ENDIF			

TP/IH/API/BV/04	Verify that parameter is set by Front End Application (single parameter)			
TP Origin	Specific			
Reference	ISO/TS 17575-2, Clause 7.2.1			
Initial Condition	A valid Instance instance1 is created.			
Stimulus and Expected Behaviour				
	Tester	Test Point		DUT
1	SetParameter (Instance = instance1, Parameter = "Parameter1", Value = "Value1")	AppEm	☐	
2		AppEm	☐	R:CEN175□52Error
3	Verify whether CEN175752Error equals to ERNoError			
4	IF verify NOT OK THEN TP failed ENDIF			
5	GetParameter (Instance = instance1, Parameter = "Parameter1")	AppEm	☐	
6		AppEm	☐	R: String
7	IF (String equals to Valu1) THEN T passed □ELSE TP failed ENDIF			

TP/IH/API/BV/05		Verify that parameter is set by Front End Application for multiple instances (different parameter names)		
TP Origin		Specific		
Reference		ISO/TS 17575-2, Clause 7.2.1		
Initial Condition		A valid Instance instance1 and instance2 are created.		
Stimulus and Expected Behaviour				
	Tester	Test Point		DUT
1	SetParameter (Instance = instance1, Parameter = "Parameter1", Value = "Value1")	AppEm	☐	
2		AppEm	☐	R: CEN175752Error
3	Verify whether CEN175752Error equals to ERNoError			
4	IF verify NOT OK THEN TP failed ENDIF			
5	GetParameter (Instance = instance1, Parameter = "Parameter1")	AppEm	☐	
6		AppEm	☐	R: String
7	IF (String equals ☐ to Value1) THEN GOTO step8 ELSE TP failed ENDIF			
8	SetParameter (Instance = instance2, Parameter = "Parameter2", Value = "Value2")	AppEm	☐	
9		AppEm	☐	R: CEN175752Error
10	Verify whether CEN175752Error equals to ERNoError			
11	IF verify NOT ☐ K THEN TP failed ENDIF			
12	GetParameter (Instance = instance2, Parameter = "Parameter2")	AppEm	☐	
13		AppEm	☐	R: String
14	IF (String equals to Value2) THEN TP passed ELSE TP failed ENDIF			

TP/IH/API/BV/06	Verify that parameter is set by Front End Application for multiple instances (the same parameter names)			
TP Origin	Specific			
Reference	ISO/TS 17575-2, Clause 7.2.1			
Initial Condition	A valid Instance instance1 and instance2 are created.			
Stimulus and Expected Behaviour				
	Tester	Test Point		DUT
1	SetParameter (Instance = instance1, Parameter = "Parameter1", Value = "Value1")	AppEm	☐	
2		AppEm	☐	R: CEN175752Error
3	Verify whether CEN175752Error equals to ERNoError			
4	IF verify NOT OK THEN TP failed ENDIF			
5	GetParameter (Instance = instance1, Parameter = "Parameter1")	AppEm	☐	
6		AppEm	☐	R: String
7	IF (String equals ☐ to Value1) THEN GOTO step8 ELSE TP failed ENDIF			
8	SetParameter (Instance = instance2, Parameter = "Parameter1", Value = "Value2")	AppEm	☐	
9		AppEm	☐	R: CEN175752Error
10	Verify whether CEN175752Error equals to ERNoError			
11	IF verify NOT O ☐ THEN TP failed ENDIF			
12	GetParameter (Instance = instance2, Parameter = "Parameter1")	AppEm	☐	
13		AppEm	☐	R: String
14	IF (String equals to Value2) THEN TP passed ELSE TP failed ENDIF			

TP/IH/API/BV/07		Verify that parameter is updated by Front End Application		
TP Origin		Specific		
Reference		ISO/TS 17575-2, Clause 7.2.1		
Initial Condition		A valid Instance instance1 is created.		
Stimulus and Expected Behaviour				
	Tester	Test Point		DUT
1	SetParameter (Instance = instance1, Parameter = "Parameter1", Value = "Value1")	AppEm	☐	
2		AppEm	☐	R: CEN175752Error
3	Verify whether CEN175752Error equals to ERNoError			
4	IF verify NOT OK THEN TP failed ENDIF			
5	GetParameter (Instance = instance1, Parameter = "Parameter1")	AppEm	☐	
6		AppEm	☐	R: String
7	IF (String equals ☐ to Value1) THEN GOTO step8 ELSE TP failed ENDIF			
8	SetParameter (Instance = instance1, Parameter = "Parameter1", Value = "Value2")	AppEm	☐	
9		AppEm	☐	R: CEN175752Error
10	Verify whether CEN175752Error equals to ERNoError			
11	IF verify NOT ☐ OK THEN TP failed ENDIF			
12	GetParameter (Instance = instance1, Parameter = "Parameter1")	AppEm	☐	
13		AppEm	☐	R: String
14	IF (String equals to Value2) THEN GOTO step8 ELSE TP failed ENDIF			

TP/IH/API/BV/08	Verify that parameter's value is fetched by the Front End Application			
TP Origin	Specific			
Reference	ISO/TS 17575-2, Clause 7.2.1			
Initial Condition	A valid Instance instance1 is created. Parameter1 has already been set with value1.			
Stimulus and Expected Behaviour				
	Tester	Test Point		DUT
1	GetParameter (Instance = instance1, Parameter = "Parameter1")	AppEm	☐	
2		AppEm	☐	R: String
3	IF (String equals to Value1) THEN TP passed ELSE TP failed ENDIF			

TP/IH/API/BV/09	Verify that parameter's value is fetched by the Front End Application (multiple instances)
TP Origin	Specific
Reference	ISO/TS 17575-2, Clause 7.2.1
Initial Condition	A valid Instance instance1, instance2 and instance3 are created. Parameter1 in instance1 has already been set with value1. Parameter2 in instance1 has already been set with value2. Parameter1 in instance3 has already been set with value3.

Stimulus and Expected Behaviour

	Tester	Test Point		DUT
1	GetParameter (Instance = instance1, Parameter = "Parameter1")	AppEm	☐	
2		AppEm	☐	R: String
3	IF (String equals to Value1) THEN GOTO step4 ELSE TP failed ENDIF			
4	GetParameter (Instance = instance2, Parameter = "Parameter2")	AppEm	☐	
5		AppEm	☐	R: String
6	IF (String equals to Value2) THEN GOTO step7 ELSE ☐ P failed ENDIF			
7	GetParameter (Instance = instance3, Parameter = "Parameter1")	AppEm	☐	
8		AppEm	☐	R: String
9	IF (String equals to Value3) THEN TP passed ELSE TP failed ENDIF			

TP/IH/API/BV/10	Verify that parameter is deleted by Front End Application (single parameter)			
TP Origin	Specific			
Reference	ISO/TS 17575-2, Clause 7.2.1			
Initial Condition	A valid Instance instance1 is created. Parameter1 has already been set by <i>Front End application</i> .			
Stimulus and Expected Behaviour				
	Tester	Test Point		DUT
1	DeleteParameter (Instance = instance1, Parameter = "Parameter1")	AppEm	☐	
2		AppEm	☐	R: CEN175752Error
3	Verify whether CEN175752Error equals to ERNoError			
4	IF verify NOT OK THEN TP failed ENDIF			
5	GetParameter (Instance = instance1, Parameter = "Parameter1")	AppEm	☐	
6		AppEm	☐	R: String
7	IF (String has invalid value) THEN TP passed ELSE TP failed ENDIF			

TP/IH/API/BV/11	Verify that parameter is deleted by Front End Application (multiple parameters)
TP Origin	Specific
Reference	ISO/TS 17575-2, Clause 7.2.1
Initial Condition	A valid Instance instance1, instance2 and instance3 are created. Parameter1 in instance1 has already been set. Parameter2 in instance2 has already been set. Parameter1 in instance3 has already been set.

Stimulus and Expected Behaviour

	Tester	Test Point		DUT
1	DeleteParameter (Instance = instance1, Parameter = "Parameter1")	AppEm	☐	
2		AppEm	☐	R: CEN175752Error
3	Verify whether CEN175752Error equals to ERNoError			
4	IF verify NOT OK THEN TP failed ENDIF			
5	GetParameter (Instance = instance1, Parameter = "Parameter1")	AppEm	☐	
6		AppEm	☐	R: String
7	IF (String has invalid value) THEN GOTO step8 ELSE TP failed ENDIF			
8	DeleteParameter (Instance = instance2, Parameter = "Parameter2")		☐	
9		AppEm	☐	R: CEN175752Error
10	Verify whether CEN175752Error equals to ERNoError	AppEm		
11	IF verify NOT OK THEN TP failed ENDIF			
12	GetParameter (Instance = instance2, Parameter = "Parameter2")		☐	
13			☐	R: String
14	IF (String has invalid value) THEN GOTO step15 ELSE TP failed ENDIF			

15	DeleteParameter (Instance = instance3, Parameter = "Parameter1")		☐	
16			☐	R: CEN175752Error
17	Verify whether CEN175752Error equals to ERNoError			
18	IF verify NOT OK THEN TP failed ENDIF			
19	GetParameter (Instance = instance3, Parameter = "Parameter1")		☐	
20			☐	R: String
21	IF (String has invalid value) THEN TP passed ELSE TP failed ENDIF			

TP/IH/API/BV/12	Verify whether StackAvail returns that communication stack is available			
TP Origin	Specific			
Reference	ISO/TS 17575-2, Clause 7.5			
Initial Condition	A valid Instance instance1 has already been created. Communication stack for instance1 is available			
Stimulus and Expected Behaviour				
	Tester	Test Point		DUT
1	StackAvail (Instance = instance1)	AppEm	☐	
2		AppEm	☐	R: Boolean
3	IF (returned ☐ TRUE THEN TP passed ELSE TP failed ENDIF			

TP/IH/API/BV/13	Verify whether StackAvail returns that communication stack is available (multiple instances)			
TP Origin	Specific			
Reference	ISO/TS 17575-2, Clause 7.5			
Initial Condition	A valid Instance instance1 and instance2 have already been created. Communication stack for instance1 and instance2 is available			
Stimulus and Expected Behaviour				
	Tester	Test Point		DUT
1	StackAvail (Instance = instance1)	AppEm	☐	
2		AppEm	☐	R: Boolean
3	IF (returned FALSE) THEN TP failed ENDIF			
4	StackAvail (Instance = instance2)	AppEm	☐	
5		AppEm	☐	R: Boolean
6	IF (returned TRUE) THEN TP passed ELSE TP failed ENDIF			

TP/IH/API/BV/14	Verify whether StackAvail returns that communication stack is unavailable			
TP Origin	Specific			
Reference	ISO/TS 17575-2, Clause 7.5			
Initial Condition	A valid Instance instance1 has already been created. Communication stack for instance1 is not available			
Stimulus and Expected Behaviour				
	Tester	Test Point		DUT
1	StackAvail (Instance = instance1)	AppEm	☐	
2		AppEm	☐	R: Boolean
3	IF (returned FALSE) THEN TP passed ELSE TP failed ENDIF			

TP/IH/API/BV/15	Verify whether StackAvail returns that communication stack is unavailable (multiple instances)			
TP Origin	Specific			
Reference	ISO/TS 17575-2, Clause 7.5			
Initial Condition	A valid Instance instance1 and instance2 have already been created. Communication stack for instance1 and instance2 are not available			
Stimulus and Expected Behaviour				
	Tester	Test Point		DUT
1	StackAvail (Instance = instance1)	AppEm	☐	
2		AppEm	☐	R: Boolean
3	IF (returned TRUE) THEN TP failed ENDIF			
4	StackAvail (Instance = instance2)	AppEm	☐	
5		AppEm	☐	R: Boolean
6	IF (returned FALSE) THEN TP passed ELSE TP failed ENDIF			

TP/IH/API/BV/16	Verify whether StackAvail returns that communication stack is available (for first instance) and unavailable (for second instance)
TP Origin	Specific
Reference	ISO/TS 17575-2, Clause 7.5
Initial Condition	A valid Instance instance1 and instance2 has already been created. Communication stack for instance1 is available. Communication stack for instance2 is unavailable

Stimulus and Expected Behaviour

	Tester	Test Point		DUT
1	StackAvail (Instance = instance1)	AppEm	☐	
2		AppEm	☐	R: Boolean
3	IF (returned FALSE) THEN TP failed ENDIF			
4	StackAvail (Instance = instance2)	AppEm	☐	
5		AppEm	☐	R: Boolean
6	IF (returned FALSE) THEN TP passed ELSE TP failed ENDIF			

TP/IH/API/BV/17	Dropping the instance with SENormal severity			
TP Origin	Specific			
Reference	ISO/TS 17575-2, Clause 7.2			
Initial Condition	A valid Instance instance1 has already been created. No session exists for instance1.			
Stimulus and Expected Behaviour				
	Tester	Test Point		DUT
1	DropInstance (Instance = instance1, Severity = SENormal)	AppEm	☐	
2		AppEm	☐	R: CEN175752Error
3	Verify whether CEN175752Error equals to ERNoError			
4	IF verify OK THEN TP passed ELSE TP failed ENDIF			

TP/IH/API/BV/18	Dropping the instance with SEUrgent severity			
TP Origin	Specific			
Reference	ISO/TS 17575-2, Clause 7.2			
Initial Condition	A valid Instance instance1 has already been created. No session exists for instance1.			
Stimulus and Expected Behaviour				
	Tester	Test Point		DUT
1	DropInstance (Instance = instance1, Severity = SEUrgent)	AppEm	☐	
2		AppEm	☐	R: CEN175752Error
3	Verify whether CEN175752Error equals to ERNoError			
4	IF verify OK THEN TP passed ELSE TP failed ENDIF			

TP/IH/API/BV/19	Dropping the instance with SEUnconditional severity			
TP Origin	Specific			
Reference	ISO/TS 17575-2, Clause 7.2			
Initial Condition	A valid Instance instance1 has already been created. No session exists for instance1.			
Stimulus and Expected Behaviour				
	Tester	Test Point		DUT
1	DropInstance (Instance = instance1, Severity = SEUnconditional)	AppEm	☐	
2		AppEm	☐	R: CEN175752Error
3	Verify whether CEN175752Error equals to ERNoError			
4	IF verify OK THEN TP passed ELSE TP failed ENDIF			

TP/IH/API/BV/20	Dropping the instance with SEUnconditional severity once session is in STStarting state			
TP Origin	Specific			
Reference	ISO/TS 17575-2, Clause 7.2			
Initial Condition	A valid Instance instance1 has already been created. Session for instance1 is being established. Session is in state Session is in state STStarting NOTE see test purposes for other operations and corresponding to this state which specify the steps how to fulfil this precondition. Correct parameterization has already been done to establish session (example: ipAddress, port, url, protocol, PDP context, etc. are set)			
Stimulus and Expected Behaviour				
	Tester	Test Point		DUT
1	DropInstance (Instance = instance1, Severity = SEUnconditional)	AppEm	☐	
2		AppEm	☐	R: CEN175752Error
3	Verify whether CEN175752Error equals to ERNoError			
4	IF verify OK THEN TP passed ELSE TP fail ☐ d ENDIF			

TP/IH/API/BV/21	Dropping the instance with SEUnconditional severity once session is in STSessionIdle state			
TP Origin	Specific			
Reference	ISO/TS 17575-2, Clause 7.2			
Initial Condition	A valid Instance instance1 has already been created. Session for instance1 is established. Session is in state Session is in state STSessionIdle NOTE see test purposes for other operations and corresponding to this state which specify the steps how to fulfil this precondition. Correct parameterization has already been done to establish session (example: ipAddress, port, url, protocol, PDP context, etc. are set)			
Stimulus and Expected Behaviour				
	Tester	Test Point		DUT
1	DropInstance (Instance = instance1, Severity = SEUnconditional)	AppEm	☐	
2		AppEm	☐	R: CEN175752Error
3	Verify whether CEN175752Error equals to ERNoError			
4	IF verify OK THEN TP passed ELSE TP failed ENDIF			

TP/IH/API/BV/22	Dropping the instance with SEUnconditional severity once session is in STSendingADU state			
TP Origin	Specific			
Reference	ISO/TS 17575-2, Clause 7.2			
Initial Condition	A valid Instance instance1 has already been created. Session for instance1 is established. Session is in state Session is in state STSendingADU NOTE see test purposes for other operations and corresponding to this state which specify the steps how to fulfil this precondition. Correct parameterization has already been done to establish session (example: ipAddress, port, url, protocol, PDP context, etc. are set)			
Stimulus and Expected Behaviour				
	Tester	Test Point		DUT
1	DropInstance (Instance = instance1, Severity = S SEUnconditional)	AppEm	☐	
2		AppEm	☐	R: CEN175752Error
3	Verify whether CEN175752Error equals to ERNoError			
4	IF verify OK THEN TP passed ELSE TP failed ENDIF			

TP/IH/API/BV/23	Dropping the instance with SEUnconditional severity once session is in STSendingADURequest state			
TP Origin	Specific			
Reference	ISO/TS 17575-2, Clause 7.2			
Initial Condition	A valid Instance instance1 has already been created. Session for instance1 is established. Session is in state Session is in state STSendingADURequest NOTE see test purposes for other operations and corresponding to this state which specify the steps how to fulfil this precondition. Correct parameterization has already been done to establish session (example: ipAddress, port, url, protocol, PDP context, etc. are set)			
Stimulus and Expected Behaviour				
	Tester	Test Point		DUT
1	DropInstance (Instance = instance1, Severity = SEUnconditional)	AppEm	☐	
2		AppEm	☐	R: CEN175752Error
3	Verify whether CEN175752Error equals to ERNoError			
4	IF verify OK THEN TP passed ESE TP failed ENDIF			

TP/IH/API/BV/24	Dropping the instance with SEUnconditional severity once session is in STSendingUnformattedADU state			
TP Origin	Specific			
Reference	ISO/TS 17575-2, Clause 7.2			
Initial Condition	A valid Instance instance1 has already been created. Session for instance1 is established. Session is in state Session is in state STSendingUnformattedADU NOTE see test purposes for other operations and corresponding to this state which specify the steps how to fulfil this precondition. Correct parameterization has already been done to establish session (example: ipAddress, port, url, protocol, PDP context, etc. are set)			
Stimulus and Expected Behaviour				
	Tester	Test Point		DUT
1	DropInstance (Instance = instance1, Severity = SEUnconditional)	AppEm	☐	
2		AppEm	☐	R: CEN175752Error
3	Verify whether CEN175752Error equals to ERNoError			
4	IF verify OK THEN TP passed ELS TP failed ENDIF			

TP/IH/API/BV/25	Dropping the instance with SEUnconditional severity once session is in STSessionRxADUs state			
TP Origin	Specific			
Reference	ISO/TS 17575-2, Clause 7.2			
Initial Condition	A valid Instance instance1 has already been created. Session for instance1 is established. Session is in state Session is in state STSessionRxADUs NOTE see test purposes for other operations and corresponding to this state which specify the steps how to fulfil this precondition. Correct parameterization has already been done to establish session (example: ipAddress, port, url, protocol, PDP context, etc. are set)			
Stimulus and Expected Behaviour				
	Tester	Test Point		DUT
1	DropInstance (Instance = instance1, Severity = SEUnconditional)	AppEm	☐	
2		AppEm	☐	R: CEN175752Error
3	Verify whether CEN175752Error equals to ERNoError			
4	IF verify OK THEN TP passed ELSE TP ☐ failed ENDIF			

TP/IH/API/BV/26	Dropping the instance with SEUnconditional severity once session is in STErrored state			
TP Origin	Specific			
Reference	ISO/TS 17575-2, Clause 7.2			
Initial Condition	A valid Instance instance1 has already been created. Session for instance1 is established. Session is in state Session is in state STErrored NOTE see test purposes for other operations and corresponding to this state which specify the steps how to fulfil this precondition. Correct parameterization has already been done to establish session (example: ipAddress, port, url, protocol, PDP context, etc. are set)			
Stimulus and Expected Behaviour				
	Tester	Test Point		DUT
1	DropInstance (Instance = instance1, Severity = SEUnconditional)	AppEm	☐	
2		AppEm	☐	R: ☐ CEN15752Error
3	Verify whether CEN175752Error equals to ERNoError			
4	IF verify OK THEN TP passed ELSE TP failed ENDI			

TP/IH/API/BV/27	Dropping the instance with SEUnconditional severity once session is in STEnding state			
TP Origin	Specific			
Reference	ISO/TS 17575-2, Clause 7.2			
Initial Condition	A valid Instance instance1 has already been created. Session for instance1 is established. Session is in state Session is in state STEnding NOTE see test purposes for other operations and corresponding to this state which specify the steps how to fulfil this precondition. Correct parameterization has already been done to establish session (example: ipAddress, port, url, protocol, PDP context, etc. are set)			
Stimulus and Expected Behaviour				
	Tester	Test Point		DUT
1	DropInstance (Instance = instance1, Severity = SEUnconditional)	AppEm	☐	
2		AppEm	☐	R: CEN175752Errr
3	Verify whether CEN175752Error equals to ERNoError			
4	IF verify OK THEN TP passed ELSE TP failed ENDIF			

A.2.2 BI test purposes (Invalid Behaviour tests)

Test subgroup objective:

- to test DUT invalid behaviour with respect to instance initialization;
- to test DUT invalid behaviour with respect to parameter setting;
- to test DUT invalid behaviour with respect to parameter getting;
- to test DUT invalid behaviour with respect to parameter deleting;
- to test DUT invalid behaviour with respect to availability of communications stack;
- to test DUT invalid behaviour with respect to dropping the session including following severities:
 - SENormal;
 - SEUrgent;

TP/IH/API/BI/01	Verify that FE Communications API returns invalid instance once FE Application selected invalid communication stack			
TP Origin	Specific			
Reference	ISO/TS 17575-2, Clause 7.2.1			
Initial Condition	Set of Callback instances is instantiated.			
Stimulus and Expected Behaviour				
	Tester	Test Point		DUT
1	InitialiseInstance (StackID =invalidStack, Callbacks = cb1)	AppEm	☐	
2		AppEm	☐	R: Instance
3	Verify whether Instance is invalid			
4	IF verify OK THEN TP passed ELSE TP failed ENDIF			

TP/IH/API/BI/02	Verify that FE Communications API returns invalid instance once FE Application provides invalid Callbacks			
TP Origin	Specific			
Reference	ISO/TS 17575-2, Clause 7.2.1			
Initial Condition	Front End Communications API must handle at least one underlying communications stacks which StackID equals to stack1.			
Stimulus and Expected Behaviour				
	Tester	Test Point		DUT
1	InitialiseInstance (StackID =invalidStack, Callbacks = ∅)	AppEm	☐	
2		AppEm	☐	R: Instance
3	Verify whether Instance is invalid			
4	IF verify OK THEN TP passed ELSE TP failed ENDIF			

TP/IH/API/BI/03	Verify parameter setting upon invalid instance			
TP Origin	Specific			
Reference	ISO/TS 17575-2, Clause 7.2.1			
Initial Condition				
Stimulus and Expected Behaviour				
	Tester	Test Point		DUT
1	SetParameter (Instance = invalidInstance1, Parameter = "Parameter1", Value = "Value1") Note: InvalidParameter1 may have too many characters what is not handled by FE Communications API	AppEm	☐	
2		AppEm	☐	R: CEN175752Error
3	Verify whether CEN175752Error equals to ERNotSet			
4	IF verify OK THEN GOTO step5 ELSE TP failed ENDIF			
5	GetParameter (Instance = instance1, Parameter = "InvalidParameter1")	AppEm	☐	
6		AppEm	☐	R: String
☐	IF (String is invalid) THEN TP passed ELSE TP failed ENDIF			

TP/IH/API/BI/04	Verify parameter setting upon invalid parameter			
TP Origin	Specific			
Reference	ISO/TS 17575-2, Clause 7.2.1			
Initial Condition	A valid Instance instance1 is created.			
Stimulus and Expected Behaviour				
	Tester	Test Point		DUT
1	SetParameter (Instance = instance1, Parameter = "InvalidParameter1", Value = "Value1") Note: InvalidParameter1 may have too many characters what is not handled by FE Communications API	AppEm	☐	
2		AppEm	☐	R: CEN175752Error
3	Verify whether CEN175752Error equals to ERNotSet			
4	IF verify NOT OK THEN TP failed ENDIF			
5	GetParameter (Instance = instance1, Parameter = "InvalidParameter1")	AppEm	☐	
6		AppEm	☐	R: String
7	IF (String ☐ s invalid) THEN TP passed ELSE TP failed ENDIF			

TP/IH/API/BI/05	Verify parameter setting upon invalid value			
TP Origin	Specific			
Reference	ISO/TS 17575-2, Clause 7.2.1			
Initial Condition	A valid Instance instance1 is created.			
Stimulus and Expected Behaviour				
	Tester	Test Point		DUT
1	SetParameter (Instance = instance1, Parameter = "Parameter1", Value = "InvalidValue1") Note: InvalidValue1 may have too many characters what is not handled by FE Communications API	AppEm	☐	
2		AppEm	☐	R: CEN175752Error
3	Verify whether CEN175752Error equals to ERNotSet			
4	IF verify NOT OK THEN TP failed ENDIF			
5	GetParameter (Instance = instance1, Parameter = "Parameter1")	AppEm	☐	
6		AppEm	☐	R: String
7	IF (String is invalid) THEN TP passed ELSE TP failed ENDIF			

TP/IH/API/BI/06	Verify getting the parameter's value upon invalid instance			
TP Origin	Specific			
Reference	ISO/TS 17575-2, Clause 7.2.1			
Initial Condition				
Stimulus and Expected Behaviour				
	Tester	Test Point		DUT
1	GetParameter (Instance = invalidInstance1, Parameter = "Parameter1")	AppEm	☐	
2		AppEm	☐	R: String
3	IF (String is invalid) THEN TP passed ELSE TP failed ENDIF			

TP/IH/API/BI/07	Verify getting the parameter's value once parameter does not exist			
TP Origin	Specific			
Reference	ISO/TS 17575-2, Clause 7.2.1			
Initial Condition	A valid Instance instance1 is created. InvalidParameter has not been set by <i>Front End application</i> .			
Stimulus and Expected Behaviour				
	Tester	Test Point		DUT
1	GetParameter (Instance = instance1, Parameter = "invalidParameter")	AppEm	☐	
2		AppEm	☐	R: String
3	IF (String is invalid) THEN TP passed ELSE TP failed ENDIF			

TP/IH/API/BI/08	Verify parameter deletion upon invalid instance			
TP Origin	Specific			
Reference	ISO/TS 17575-2, Clause 7.2.1			
Initial Condition				
Stimulus and Expected Behaviour				
	Tester	Test Point		DUT
1	DeleteParameter (Instance = invalidInstance1, Parameter = "Parameter1")	AppEm	☐	
2		AppEm	☐	R: CEN175752Error
3	Verify whether CEN175752Error equals to ERNotSet			
4	IF verify OK THEN TP passed ELSE TP failed ENDIF			

TP/IH/API/BI/09	Verify parameter deletion once parameter does not exist			
TP Origin	Specific			
Reference	ISO/TS 17575-2, Clause 7.2.1			
Initial Condition	A valid Instance instance1 is created.			
Stimulus and Expected Behaviour				
	Tester	Test Point		DUT
1	DeleteParameter (Instance =instance1, Parameter = "Parameter1")	AppEm	☐	
2		AppEm	☐	R: CEN175752Error
3	Verify whether CEN175752Error equals to ERNotSet			
4	IF verify OK THEN TP passed ELSE TP failed ENDIF			

TP/IH/API/BI/10	Verify whether StackAvail returns false not having instance initialized			
TP Origin	Specific			
Reference	ISO/TS 17575-2, Clause 7.5			
Initial Condition	FE Communications API is initialized			
Stimulus and Expected Behaviour				
	Tester	Test Point		DUT
1	StackAvail (Instance = invalidInstance)	AppEm	☐	
2		AppEm	☐	R: Boolean
3	IF (returned FALSE) THEN TP passed ELSE TP failed ENDIF			

TP/IH/API/BI/11	Dropping the instance with SENormal severity once no instance is initialized			
TP Origin	Specific			
Reference	ISO/TS 17575-2, Clause 7.2			
Initial Condition	FE Communications API is initialized			
Stimulus and Expected Behaviour				
	Tester	Test Point		DUT
1	DropInstance (Instance = invalidInstance, Severity = SENormal)	AppEm	☐	
2		AppEm	☐	R: CEN175752Error
3	Verify whether CEN175752Error equals ☐ o EUnknownInstance			
4	IF verify OK THEN TP passed ELSE TP failed ENDIF			

TP/IH/API/BI/12	Dropping the instance with SEUrgent severity once no instance is initialized			
TP Origin	Specific			
Reference	ISO/TS 17575-2, Clause 7.2			
Initial Condition	FE Communications API is initialized			
Stimulus and Expected Behaviour				
	Tester	Test Point		DUT
1	DropInstance (Instance = invalidInstance, Severity = SEUrgent)	AppEm	☐	
2		AppEm	☐	R: CEN175752Error
3	Verify whether CEN175752Error equals to ERUnknownInstance			
4	IF verify OK THEN TP passed ELSE TP failed ENDIF			

TP/IH/API/BI/13	Dropping the instance with SEUnconditional severity once no instance is initialized			
TP Origin	Specific			
Reference	ISO/TS 17575-2, Clause 7.2			
Initial Condition	FE Communications API is initialized			
Stimulus and Expected Behaviour				
	Tester	Test Point		DUT
1	DropInstance (Instance = invalidInstance, Severity = SEUnconditional)	AppEm	☐	
2		AppEm	☐	R: CEN175752Error
3	Verify whether CEN175752Error equals to ERUnknownInstance			
4	IF verify OK THEN TP passed ELSE TP failed ENDIF			

TP/IH/API/BI/14	Dropping the instance with SENormal severity once session is in STStarting state			
TP Origin	Specific			
Reference	ISO/TS 17575-2, Clause 7.2			
Initial Condition	A valid Instance instance1 has already been created. Session for instance1 is being established. Session is in state Session is in state STStarting NOTE see test purposes for other operations and corresponding to this state which specify the steps how to fulfil this precondition. Correct parameterization has already been done to establish session (example: ipAddress, port, url, protocol, PDP context, etc. are set)			
Stimulus and Expected Behaviour				
	Tester	Test Point		DUT
1	DropInstance (Instance = instance1, Severity = SENormal)	AppEm	☐	
2		AppEm	☐	R: CEN175752Error
3	Verify whether CEN175752Error equals to ERBadState			
4	IF verify OK THEN TP passed ELSE TP failed ENDF			

TP/IH/API/BI/15	Dropping the instance with SENormal severity once session is in STSessionIdle state			
TP Origin	Specific			
Reference	ISO/TS 17575-2, Clause 7.2			
Initial Condition	A valid Instance instance1 has already been created. Session for instance1 is established. Session is in state Session is in state STSessionIdle NOTE see test purposes for other operations and corresponding to this state which specify the steps how to fulfil this precondition. Correct parameterization has already been done to establish session (example: ipAddress, port, url, protocol, PDP context, etc. are set)			
Stimulus and Expected Behaviour				
	Tester	Test Point		DUT
1	DropInstance (Instance = instance1, Severity = SENormal)	AppEm	☐	
2		AppEm	☐	R: CEN175752 Err ☐ r
3	Verify whether CEN175752Error equals to ERBadState			
4	IF verify OK THEN TP passed ELSE TP failed ENDIF			