ETSI TS 151 013 V5.0.1 (2003-06)

Technical Specification

Digital cellular telecommunications system (Phase 2+); Test specification for Subscriber Identity Module (SIM) Application Programming Interface (API) for Java Card (3GPP TS 51.013 version 5.0.1 Release 5)



Reference
RTS/TSGT-0351013v501

Keywords
GSM

ETSI

650 Route des Lucioles F-06921 Sophia Antipolis Cedex - FRANCE

Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 - NAF 742 C Association à but non lucratif enregistrée à la Sous-Préfecture de Grasse (06) N° 7803/88

Important notice

Individual copies of the present document can be downloaded from: <u>http://www.etsi.org</u>

The present document may be made available in more than one electronic version or in print. In any case of existing or perceived difference in contents between such versions, the reference version is the Portable Document Format (PDF). In case of dispute, the reference shall be the printing on ETSI printers of the PDF version kept on a specific network drive within ETSI Secretariat.

Users of the present document should be aware that the document may be subject to revision or change of status.

Information on the current status of this and other ETSI documents is available at

http://portal.etsi.org/tb/status/status.asp

If you find errors in the present document, send your comment to: editor@etsi.org

Copyright Notification

No part may be reproduced except as authorized by written permission. The copyright and the foregoing restriction extend to reproduction in all media.

© European Telecommunications Standards Institute 2003. All rights reserved.

DECTTM, **PLUGTESTS**TM and **UMTS**TM are Trade Marks of ETSI registered for the benefit of its Members. **TIPHON**TM and the **TIPHON logo** are Trade Marks currently being registered by ETSI for the benefit of its Members. **3GPP**TM is a Trade Mark of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners.

Intellectual Property Rights

IPRs essential or potentially essential to the present document may have been declared to ETSI. The information pertaining to these essential IPRs, if any, is publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: "Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards", which is available from the ETSI Secretariat. Latest updates are available on the ETSI Web server (http://webapp.etsi.org/IPR/home.asp).

Pursuant to the ETSI IPR Policy, no investigation, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

Foreword

This Technical Specification (TS) has been produced by ETSI 3rd Generation Partnership Project (3GPP).

The present document may refer to technical specifications or reports using their 3GPP identities, UMTS identities or GSM identities. These should be interpreted as being references to the corresponding ETSI deliverables.

The cross reference between GSM, UMTS, 3GPP and ETSI identities can be found under http://webapp.etsi.org/key/queryform.asp.

Contents

Intelle	ectual Property Rights	2
Forew	ord	2
Forew	ord	9
1	Scope	10
2	References	10
3	Definitions and abbreviations	11
3.1	Definitions	11
3.2	Abbreviations	11
4	Test Environment	
4.1	Applicability	
4.2	Test environment description	
4.3	Tests format	
4.3.1	Test Area Reference	
4.3.1.1	1	
4.3.1.2		
4.3.1.3		
4.3.1.4		
4.4	Initial Conditions	
4.5	Package name	
4.6	AID Coding	
4.6.1	Specific Test Applet Name for API	
4.6.2 4.7	Specific Test Applet Name for Framework	
4.7.1	APDU tool	
4.7.1	Util package	
4.7.3	Applet installation parameters	
4.7.3.1		
4.7.3.2	√ 1	
4.8	Testing methodology	
4.8.1	Test interfaces and facilities.	
5	Test plan	18
6	API Test Plan	19
6.1	Package sim.access:	
6.1.1	Interface SIMView	
6.1.1.1	Constants	19
6.1.1.2	Method select(short fid, byte[] fci, short fciOffset, short fciLength)	19
6.1.1.3	Method select (short fid)	23
6.1.1.4	Method status	25
6.1.1.5	Method readBinary	27
6.1.1.6	1 •	
6.1.1.7		
6.1.1.8	1	
6.1.1.9		
6.1.1.1		
6.1.1.1		
6.1.1.1		
6.1.2	Class SIMSystem	
6.1.2.1		
6.1.3	Class SIMViewException	
6.1.3.1		
6.1.3.2		
6.1.3.3	Reason Codes	58

6.2	Package sim.toolkit	58
6.2.1	Interface ToolkitConstants	
6.2.1.1	Constants	58
6.2.2	Interface ToolkitInterface	59
6.2.2.1	Method processToolkit	59
6.2.3	Class EditHandler	60
6.2.4	Class EnvelopeHandler	60
6.2.4.1	Method getEnvelopeTag	60
6.2.4.2	Method getItemIdentifier	61
6.2.4.3	Method getSecuredDataLength	62
6.2.4.4	Method getSecuredDataOffset	
6.2.4.5	Method getTheHandler	
6.2.4.6	Method getTPUDLOffset	
6.2.4.7	Method getLength	
6.2.4.8	Method copy	
6.2.4.9	Method findTLV	
6.2.4.10	Method getValueLength	
6.2.4.11	Method getValueByte	
6.2.4.12	Method copyValue	
6.2.4.13	Method compareValue	
6.2.4.14	Method findAndCopyValue(byte tag, byte[] dstBuffer, short dstOffset)	84
6.2.4.15	Method findAndCopyValue(byte tag, byte occurrence, short valueOffset, byte[] dstBuffer, short	0.4
6.2.4.16	dstOffset, short dstLength)	
6.2.4.17	Method findAndCompareValue(byte tag, byte occurrence, short valueOffset, byte[]	0>
0.2.4.17	compareBuffer, short compareOffset, short compareLength)	0.1
6.2.4.18	Method getCapacity	
6.2.4.19	Method getUserDataLength	
6.2.4.19	Method getChannelIdentifier	
6.2.5	Class EnvelopeResponseHandler	
6.2.5.1	Method getTheHandler	
6.2.5.2	Method post	
6.2.5.3	Method postAsBERTLV	
6.2.5.4	Method getLength	
6.2.5.5	Method copy	
6.2.5.6	Method findTLV	
6.2.5.7	Method getValueLength	
6.2.5.8	Method getValueByte	
6.2.5.9	Method copyValue	
6.2.5.10	Method compare Value	
6.2.5.11	Method findAndCopyValue(byte tag, byte[] dstBuffer, short valueOffset)	
6.2.5.12	Method findAndCopyValue(byte tag, byte occurrence, short valueOffset, byte[] dstBuffer, short	
	dstOffset, short dstLength)	
6.2.5.13	Method findAndCompareValue(byte tag, byte[] compareBuffer, short compareOffset)	.121
6.2.5.14	Method findAndCompareValue(byte tag, byte occurrence, short valueOffset, byte[]	
	compareBuffer, short compareOffset, short compareLength)	
6.2.5.15	Method appendArray	
6.2.5.16	Method appendTLV(byte tag, byte value)	
6.2.5.17	Method appendTLV(byte tag, byte value1, byte value2)	
6.2.5.18	Method appendTLV(byte tag, byte[] value, short valueoffset, short valuelength)	
6.2.5.19	Method appendTLV(byte tag, byte value1, byte[] value2, short value2offset, short value2length).	
6.2.5.20	Method clear	
6.2.5.21	Method getCapacity	
6.2.6	Class MEProfile	
6.2.6.1	Method check (byte index)	
6.2.6.2 6.2.6.3	Method check (byte [] mask, short offset, short length)	
6.2.6.4	Method check (short index)	
6.2.6.4	Method copy (short startOffset, byte[] dstBuffer, short dstOffset, short dstLength)	
6.2.7	Class ProactiveHandler	
6.2.7.1	Method getTheHandler	
6272		145

6.2.7.3	Method initDisplayText	
6.2.7.4	Method initGetInkey	
6.2.7.5	Method initGetInput	
6.2.7.6	Method send	
6.2.7.7	Method getLength	
6.2.7.8	Method copy	
6.2.7.9	Method findTLV	
6.2.7.10	Method getValueLength	
6.2.7.11	Method getValueByte	
6.2.7.12	Method copyValue	
6.2.7.13	Method compareValue	
6.2.7.14	Method findAndCopyValue(byte tag, byte[] dstBuffer, short valueOffset)	171
6.2.7.15	Method findAndCopyValue(byte tag, byte occurence, short valueOffset, byte[] dstBuffer, short	
	dstOffset, short dstLength)	
6.2.7.16	Method findAndCompareValue(byte tag, byte[] compareBuffer, short compareOffset)	176
6.2.7.17	Method findAndCompareValue(byte tag, byte occurence, short valueOffset, byte[]	
	compareBuffer, short compareOffset, short compareLength)	
6.2.7.18	Method appendArray	
6.2.7.19	Method appendTLV(byte tag, byte value)	
6.2.7.20	Method appendTLV(byte tag, byte value1, byte value2)	
6.2.7.21	Method appendTLV(byte tag, byte[] value, short valueoffset, short valuelength)	
6.2.7.22	Method appendTLV(byte tag, byte value1, byte[] value2, short value2offset, short value2length)	
6.2.7.23	Method clear	
6.2.7.24	Method getCapacity	
6.2.7.25	Method initCloseChannel	
6.2.8	Class ProactiveResponseHandler	
6.2.8.1	Method copyAdditionalInformation	
6.2.8.2	Method copyTextString	
6.2.8.3	Method getAdditionalInformationLength	
6.2.8.4	Method getGeneralResult	
6.2.8.5	Method getItemIdentifier	
6.2.8.6	Method getTextStringCodingScheme	
6.2.8.7	Method GetTextStringLength	
6.2.8.8	Method getTheHandler	
6.2.8.9	Method getLength	
6.2.8.10	Method copy	
6.2.8.11	Method findTLV	
6.2.8.12	Method getValueLength	
6.2.8.13	Method getValueByte	
6.2.8.14	Method copyValue	
6.2.8.15	Method compareValue	
6.2.8.16	Method findAndCopyValue(byte tag, byte[] dstBuffer, short valueOffset)	226
6.2.8.17	Method findAndCopyValue(byte tag, byte occurence, short valueOffset, byte[] dstBuffer, short	220
60 010	dstOffset, short dstLength)	
6.2.8.18	Method findAndCompareValue(byte tag, byte[] compareBuffer, short compareOffset)	232
6.2.8.19	Method findAndCompareValue(byte tag, byte occurence, short valueOffset, byte[]	224
< 0.000	compareBuffer, short compareOffset, short compareLength)	
6.2.8.20	Method getCapacity	
6.2.8.21	Method getChannelIdentifier	
6.2.8.22	Method copyChannelData	
6.2.9	Class ToolkitRegistry	
6.2.9.1	Method allocateTimer	
6.2.9.2	Method changeMenuEntry	
6.2.9.3	Method clearEvent	
6.2.9.4	Method disableMenuEntry	
6.2.9.5	Method enableMenuEntry	
6.2.9.6	Method getPallInterval	
6.2.9.7	Method getPollInterval	
6.2.9.8	Method intMenuEntry	
6.2.9.9	Method releases Timer	
6.2.9.10	Method requestPollInterval	
6.2.9.11	Method requestPollInterval	∠/\

6.2.9.12	Method setEvent	
6.2.9.13	Method setEventList	
6.2.10	Class ViewHandler	
6.2.11	Class ToolkitException	
6.2.11.1	Exception Constants	
6.2.11.2	Constructor ToolkitException	
6.2.11.3	Method throwIt	
6.3	SIM Toolkit Framework	
6.3.1	Minimum Handler Availability	
6.3.1.1	ProactiveHandler	
6.3.1.2	ProactiveResponseHandler	
6.3.1.3	EnvelopeHandler	
6.3.1.4	EnvelopeResponseHandler	
6.3.2	Handler Integrity	
6.3.2.1	ProactiveHandler	
6.3.2.2	ProactiveResponseHandler	
6.3.2.3	EnvelopeHandler	
6.3.2.4	EnvelopeResponseHandler	
6.3.3	Applet Triggering	
6.3.3.1	EVENT_PROFILE_DOWNLOAD	
6.3.3.2	EVENT_MENU_SELECTION	
6.3.3.3	EVENT_MENU_SELECTION_HELP_REQUEST	
6.3.3.4	EVENT_FORMATTED_SMS_PP_ENV	
6.3.3.5	EVENT_UNFORMATTED_SMS_PP_ENV	
6.3.3.6	EVENT_CALL_CONTROL_BY_SIM	
6.3.3.7	EVENT_MO_SHORT_MESSAGE_CONTROL_BY_SIM	
6.3.3.8	EVENT_TIMER_EXPIRATION	
6.3.3.9	EVENT_UNFORMATTED_SMS_CB	
6.3.3.10	EVENT_EVENT_DOWNLOAD_MT_CALL	
6.3.3.11	EVENT_EVENT_DOWNLOAD_CALL_CONNECTED	
6.3.3.12	EVENT_EVENT_DOWNLOAD_CALL_DISCONNECTED	
6.3.3.13	EVENT_EVENT_DOWNLOAD_LOCATION_STATUS	
6.3.3.14	EVENT_EVENT_DOWNLOAD_USER_ACTIVITY	
6.3.3.15 6.3.3.16	EVENT_EVENT_DOWNLOAD_IDLE_SCREEN_AVAILABLE EVENT_EVENT_DOWNLOAD_CARD_READER_STATUS	
6.3.3.17	EVENT_EVENT_DOWNLOAD_CARD_READER_STATUS	
6.3.3.18	EVENT_UNRECOGNIZED_EN VELOPE EVENT_STATUS_COMMAND	
6.3.3.19	EVENT_STATUS_COMMAND	
6.3.3.20	EVENT_FORMATTED_SMS_CBEVENT_EVENT_BOWNLOAD_LANGUAGE_SELECTION	
6.3.3.21	EVENT_EVENT_DOWNLOAD_BROWSER_TERMINATION	
6.3.3.22	EVENT_EVENT_BOWNLOAD_BROWSER_TERMINATION	
6.3.3.23	EVENT_EVENT_DOWNLOAD_DATA_AVAILABLE	
6.3.3.24	EVENT_EVENT_DOWNLOAD_CHANNEL_STATUS	
6.3.3.25	EVENT_FORMATTED_SMS_PP_UPD	
6.3.3.26	EVENT_UNFORMATTED_SMS_PP_UPD	
6.3.4	Proactive Command Sending by the STF	
6.3.4.1	System Proactive Commands	
6.3.4.2	Interaction with GSM commands	
6.3.4.3	Proactive Command Control	
6.3.5	Exception Handling	
6.3.5.1	Hide Exceptions from the ME	
6.3.5.2	Interaction with Multiple Triggering	
6.3.6	Framework Security Management	
6.3.6.1	Input Data	
6.3.6.2	Output Data	
6.3.7	Envelope Response Posting	
6.3.7.1	EVENT_CALL_CONTROL_BY_SIM	
6.3.7.2	EVENT_MO_SHORT_MESSAGE_CONTROL_BY_SIM	
6.3.7.3	EVENT_UNRECOGNIZED_ENVELOPE	
6.3.7.4	EVENT_FORMATTED_SMS_PP_ENV	
6.3.8	Toolkit Installation	425
6.3.8.1	Timers Allocation	425

6.3.8.		
6.3.8.		
6.3.8.		
6.3.8.		
6.3.8.		
6.3.8.	→	
6.3.8.	· · · · · · · · · · · · · · · · · · ·	
6.3.8.	· · · · · · · · · · · · · · · · · · ·	
6.3.9 6.3.9.	•	
6.3.9.		
6.3.9.	·	
6.3.10		
6.3.10		
6.3.10		
6.3.10		
6.3.11	11	
6.3.11		
6.3.11	1.2 Test Suite Files	455
6.3.11	1.4 Test Coverage	457
Anne	ex A (normative): Class and Methods AID numbering and acronyms	458
A.1	Sim.access	458
A.1.1		
A.1.2		
A.1.3	•	
	•	
A.2	Sim.toolkit	
A.2.1		
A.2.2		
A.2.3		
A.2.4	——····································	
A.2.5 A.2.6	1 1	
A.2.7		
A.2.7 A.2.8		
A.2.9	•	
A.2.1	ě ·	
A.2.1		
11.2.1	1 TOOKILEACOPTOII IIICUIOUS	
Anne	ex B (normative): Script file syntax and format description	46 4
D 1	Contant description	1.64
B.1	Syntax description	404
B.2	Semantics	465
D 2	E1	1.05
B.3	Example	403
B.4	Style and formatting	466
	,	
Anne	ex C (normative): Default Prepersonalization	467
C.1	General Default Prepersonalization	167
C.1	General Default Flepersonanzation	40/
C.2	Sim.Access.SimView test default prepersonalization	468
C.2.1		
C.2.2		
C.2.3	11.0	
C.2.4	inte ()	
C.2.5	Critic ()	
C.2.6		
C.2.7	crite ()	
C.2.8		
C.2.9	EF _{CNRH} (Cyclic Never Rehabilitate)	470

C.2.10	0 EF _{CARU} (Cyclic Always Read and Update)	470
C.2.1	EATR	
C.2.12	ENC (
C.2.13		
C.2.14	CHAIL ()	
C.2.15	TRAC (T	
C.2.16		
C.2.17 C.2.18	cinic ()	
C.2.19		
	ex D (normative): sim.test.util package and loading, testing and cleaning script exan	
Anne	ex E (normative): Test Area files	475
Anne	ex F (normative): AID numbering and acronyms for Framework tests	476
F.1	Toolkit Installation Parameters (TIN)	476
F.2	Minimum Handler Availability (MHA)	476
F.3	Handler Integrity (HIN)	
F.4	Applet Triggering (APT)	
F.5	Proactive Command Sending (PCS)	
F.6	Envelope Response Posting (ERP)	
F.7	Framework Security (FWS)	
F.8	File System Context (FSC)	
F.9	Exception Handling (EXH)	
F.10	Other parts transferred to framework from API (API)	
F.11	Concatenation processing (PROC)	478
Anne	ex G (normative): Configuration Parameters File	479
G.1	Syntax	479
G.2	File Contents and Organization	480
	Default values, order and processing	480
G.2.2		
G.2.3	· /	
G.2.4		
G.2.5		
G.3	Full example	
	ex H (informative): Change history	
Histo	ory	484

Foreword

This Technical Specification (TS) has been produced by the 3rd Generation Partnership Project (3GPP).

The contents of the present document are subject to continuing work within the TSG and may change following formal TSG approval. Should the TSG modify the contents of the present document, it will be re-released by the TSG with an identifying change of release date and an increase in version number as follows:

Version x.y.z

where:

- x the first digit:
 - 1 presented to TSG for information;
 - 2 presented to TSG for approval;
 - 3 or greater indicates TSG approved document under change control.
- y the second digit is incremented for all changes of substance, i.e. technical enhancements, corrections, updates, etc.
- z the third digit is incremented when editorial only changes have been incorporated in the document.

1 Scope

The present document covers the minimum characteristics considered necessary in order to provide compliance to 3GPP TS 43.019 [7].

The present document describes the technical characteristics and methods of test for testing the SIM API for Java CardTM (3GPP TS 43.019 [7]) implemented in the Subscriber Identity Modules (SIMs) for GSM. It specifies the following parts:

- test applicability;
- test environment description;
- tests format;
- test area reference;
- conformance requirements;
- test auite files;
- test procedure;
- test coverage; and
- a description of the associated testing tools that shall be used.

2 References

The following documents contain provisions which, through reference in this text, constitute provisions of the present document.

- References are either specific (identified by date of publication, edition number, version number, etc.) or non-specific.
- For a specific reference, subsequent revisions do not apply.
- For a non-specific reference, the latest version applies. In the case of a reference to a 3GPP document (including a GSM document), a non-specific reference implicitly refers to the latest version of that document *in the same Release as the present document*.
- [1] Void.
- [2] Void.
- [3] 3GPP TS 51.011: "Specification of the Subscriber Identity Module Mobile Equipment (SIM ME) interface".
- [4] 3GPP TS 11.14: "Specification of the Subscriber Identity Module Mobile Equipment (SIM ME) interface".
- [5] 3GPP TS 11.17: "Digital cellular telecommunications system (Phase 2+); Subscriber Identity Module (SIM) test specification".
- [6] Void.
- [7] 3GPP TS 43.019: "Subscriber Identity Module Application Programming Interface (SIM API) for Java CardTM; Stage 2 (Release 5)".
- [8] 3GPP TS 23.048: "Security Mechanisms for the (U)SIM application toolkit; Stage 2 (Release 5)".

[9]	ISO/IEC 7816-3 (1997): "Information technology - Identification cards - Integrated circuit(s) cards with contacts - Part 3: Electronic signals and transmission protocols".
[10]	3GPP TS 42.019: "Subscriber Identity Module Application Programming Interface (SIM API); Stage 1".
[11]	SUN Java Card Specification "Java Card 2.1 API Specification".
[12]	SUN Java Card Specification "Java Card 2.1 Runtime Environment Specification".
[13]	SUN Java Card Specification "Java Card 2.1 VM Architecture Specification".
NOTE:	SUN Java Card Specifications can be downloaded at http://java.sun.com/products/javacard .
[14]	ETSI TS 101 220: "Smart Cards; ETSI numbering system for telecommunication application providers".
[15]	3GPP TS 11.10-1: "Mobile Station (MS) conformance specification; Part 1: Conformance specification".

3 Definitions and abbreviations

3.1 Definitions

For the purposes of the present document, the terms and definitions given in 3GPP TS 51.010-1 [15] and the following apply:

applet: application built up using a number of classes which will run under the control of the Java Card virtual machine **applet installation parameters:** default values for applet installation parameters

applet loading script: file containing the APDU commands that will load and install the test applet in the card

CleanUp Script file: file containing the APDU commands that will restore the Default Initial Conditions on the SIM

Conformance Requirement Reference: description of the expected card behaviour according to 3GPP TS 43.019

expected state: state in which the SIM is supposed to be after the execution of the test procedure applied on the relevant initial conditions

security parameters: minimum security requirements defined for the applet installation process

test area: set of Test Cases applicable to a specific part (class method, framework behaviour, ...) of the 3GPP TS 43.019.

test case: elementary test that checks for compliance with one or more Conformance Requirement References **test Output file:** TBD.

test procedure: the sequence of actions/commands to perform all the test cases defined in a test area

test script file: file containing the APDU commands that will execute and verify the test results

Test Toolkit Applet: applet designed to test a specific functionality of the SIM API (3GPP TS 43.019)

3.2 Abbreviations

For the purpose of the present document, the abbreviations given in GSM 01.04 [2] and the following apply:

AC Application Code
AID Application Identifier

APDU Application Protocol Data Unit

API	Application	Programming 1	Interface

CAD Card Acceptance Device

CRR Conformance Requirements Reference

CRRC Conformance Requirement Reference Context Error CRRN Conformance Requirement Reference Normal

CRRP Conformance Requirement Reference Parameter Error

FFS For Further Study
IFD Interface Device

JCRE Java CardTM Run Time Environment

JVM Java Virtual Machine SE Sending Entity

SIM Subscriber Identity Module

4 Test Environment

This clause specifies requirements that shall be met and the testing rules that shall be followed during the test procedure.

4.1 Applicability

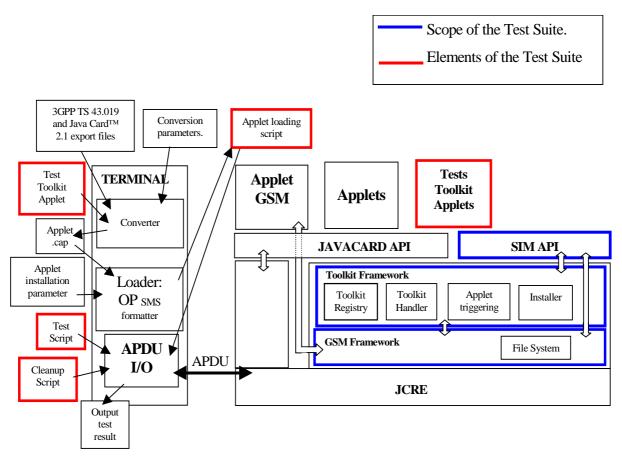
The tests defined in the present document shall be performed taking into account the services supported by the card as specified in the EF_{SST} file.

The test defined in the present document are applicable to cards implementing 3GPP TS 43.019 [7] unless otherwise stated.

The tests defined in the present document require that the card support the concatenation process with 2 concatenated SMS. Therefore the envelope handler shall support 280 bytes of data.

4.2 Test environment description

The general architecture for the test environment is.



NOTE: Figure 4.2 shows the test architecture required to test interoperability at both API and bytcode level. The latter is currently not included in the current specification. The diagram is for information.

Figure 4.2

4.3 Tests format

4.3.1 Test Area Reference

Each test area is referenced as follows:

API Testing:: 'API_[package name]_[classname]_[methodname]' where

package name:

sim.access package: '1'

sim.toolkit package: '2'

class name:

yyy: 3 letters for each class.

See Annex A for full classes acronyms list.

method name:

zzzz[input parameters]:

See Annex A for full methods name acronyms list.

FWK: framework testing

Chapter name:

```
xxx: 3 letters for each chapter

See annex F for full chapter acronyms list

Subchapter name

yyyy: : 4 letters for each subchapter

See annex F for full subchapter acronyms list

LDR: loader testing
```

4.3.1.1 Conformance requirements

The conformance requirements are expressed in the following way:

- Method prototype as listed in 3GPP TS 43.019 [7].
- Normal execution:

[TBD]

- Contains normal execution and correct parameters limit values, each referenced as a Conformance Requirement Reference Normal (CRRN).
- Parameters error:
 - Contains parameter errors and incorrect parameter limit values, each referenced as a Conformance Requirement Reference Parameter Error (CRRP).
- Context error:
 - Contains errors due to the context the method is used in, each referenced as a Conformance Requirement Reference Context Error (CRRC).

4.3.1.2 Test Area files

The files included in the Test Area use the following naming convention:

Test Script: [Test Area Reference]_[Test script number].scr
 Test Applet: [Test Area Reference]_[Test applet number].java
 Load Script: [Test Area Reference]_[Load Script number].ldr
 Cleanup Script: [Test Area Reference]_[Cleanup Script number].clr
 Parameter File: [Test Area Reference]_[Parameter File number].par

The test script, applet, installation parameters, load script, cleanup script and conversion parameters numbers start from '1'.

The test script, load script and cleanup script shall share a common syntax and format (see Annex B).

The parameter file has an own syntax (see annex G) and contains parameters to be used for CAP-file conversion and loading/cleanup script generation.

Scripts file shall be run in the following order:

```
[Test Area Reference]_1.ldr
[Test Area Reference]_1.scr
[Test Area Reference]_1.clr
[Test Area Reference]_2.ldr
```

```
[Test Area Reference]_2.scr

[Test Area Reference]_2.clr

....

[Test Area Reference]_n.ldr

[Test Area Reference]_n.scr

[Test Area Reference]_n.clr
```

In case that one of the files is not needed, it shall be skipped during the tests execution.

4.3.1.3 Test Procedure

Each test procedure contains a table to indicate the expected responses form the API and/or the APDU level as follows:

	Test Case														
ld	Description	API Expectation	APDU Expectation												
	Test Case detailed description	•	Expected response at APDU level.												

4.3.1.4 Test Coverage

The table at the end of each test procedure indicates the correspondence between the Conformance Requirements Reference (CRR) and the different test cases.

4.4 Initial Conditions

The Initial Conditions are a set of general prerequisites for the SIM prior to the execution of testing. For each test procedure described in the present document, the following rules apply to the Initial Conditions:

- unless otherwise stated, the file system and the files' content shall fulfil the requirements described in annex C;
- unless otherwise stated, before installing the applet(s) relevant to the current test procedure, all packages specific to other test procedures shall not be present.

When both statements apply, a test procedure is said to be in the "Default Initial Conditions" state.

4.5 Package name

Java packages integrating this Test Suite shall follow this naming convention:

sim.test.access.[Test Area Reference]: Java Card packages containing Test Area References for the 3GPP TS 43.019 [7] sim.access package.

sim.test.framework.[Test Area Reference]: Java Card packages containing Test Area References for the 3GPP TS 43.019 [7] framework.

sim.test.util: for the Test util package defined in this Test Suite.

sim.test.toolkit.[Test Area Reference]: Java Card packages containing Test Area References for the 3GPP TS 43.019 [7] sim.toolkit package.

EXAMPLE: The package ../sim.test.access.[Test Area Reference] creates the following directory structure ../sim/test/access/[Test Area Reference]/API_1_..._[1..n].*, where 'API_1_..._[1..n].*' are the different test applets Java source files used in [Test Area Reference].

4.6 AID Coding

The AID coding for the Test Packages, Applet classes and Applet shall be as specified in TS 101 220 [14]. In addition, the following TAR values are defined for use within the present document:

TAR Coding (3 bytes/ 24 bits):

b1	þ	2	b3	b4	k	b5	b6		b21	b2:	2	b23	þ	24	
															Specific Test Applet Name
															Test Package Identifier
															rest Package Identifier

Test package Identifier(bits b1-b3):

000 reserved (as TAR= '00.00.00' is reserved for Card Manager)

001 API

010 Framework

011 Loader

111 sim.test.util

other values are RFU

Application Provider specific data (1 byte):

'00' for Package

'01' for Applet class

'02' for Applet Instance

EXAMPLE: The AID of Package sim.test.util is 'A0 00 00 00 09 00 02 FF FF FF FF 89 E0 00 00 00'.

4.6.1 Specific Test Applet Name for API

Specific applet test name (bits b4-b24):

b4	b5	b6	b	7 k	8	b9	b.	10 b	11 b	12 b	13 b	14 b	15 b	16 b	17 b	18 b	19 b	20 b	21 b	22 }	o23 b24	1	
																							instance Number Class Number
																	<u> </u>		<u> </u>			Method	Class Number
																						Class	
																						API Tes	st Package

for API Test Package(3 bits)

001 sim.access

010 sim.toolkit

other are RFU

Class (5 bits): need to be assigned specification order see Annex A for the full list

Method (6 bits): need to be assigned specification order see Annex A for the full list

Applet Class Number (5 bits): linked to Test Area, it shall start with 1 for classes and shall be 0 for package.

Applet Instance Number (2 bits) defined in the test procedure it shall start with 01 for applet instance and shall be 00 for package and class.

4.6.2 Specific Test Applet Name for Framework

Specific applet test name (bits b4-b24):

b4	b!	5 b	6	b7	b8	b) b:	10 b	11 b	12 b	13 b	14 b	15 k	16 k	o17 k	18	b19	b20	b2	1 b2	22 b	23 1	b24	
																								RFU (set to 0)
																								Applet instance Number
																		_						Applet Class Number
																								Test Area within the chapter
										•														Chapter

for Chapter (5 bits)

00001 Toolkit Installation Parameters

00010 Minimum Handler Availability

00011 Handler Integrity

00100 Applet Triggering

00101 Proactive Command Sending

00110 Framework Security

00111 Envelope Response Posting

01000 File System Context

01001 Exception Handling

01010 Other parts transferred to framework from API

01011 Concatenation processing

other are RFU

Test Area within the chapter (6 bits): values are defined in Annex F

Applet Class number (5 bits): linked to Test Area, it shall start with 1 for classes and shall be 0 for package.

Applet Instance number (3 bits) defined in the test procedure it shall start with 01 for applet instance and shall be 00 for package and class.

4.7 Test Equipment

These subclauses recommend a minimum specification for each of the items of test equipment referenced in the tests.

4.7.1 APDU tool

This test tool shall meet the following requirements:

- be able to send command to the card TPDU;
- be able to check none, only a part, or all of the data returned;
- be able to check none, only part, or all of the status returned;
- be able to accept all valid status codes returned;
- be able to support Reader commands;
- be able to generate a log file for each test execution.
- if more data is returned than defined in the test specification, the tool shall continue;

- if less data is returned than defined in the test specification, the tool shall aborts and return an error;
- if there is an error in data or status returned, the tool shall abort and return an error.

The log file produced by the test tool shall include the following information:

- all commands issued;
- all data returned;
- all status returned;
- all errors codes;
- expected data and status in case of error;
- comments from the scripts;
- a log message to report success or failure of the test.

4.7.2 Util package

Annex D includes java source code for the sim.test.util package as well as loading, testing and cleaning script examples.

4.7.3 Applet installation parameters

4.7.3.1 Security parameters

Loading scripts shall use the following security parameters as stated in 3GPP TS 23.048 [8] for applet installation:

Parameter	Value in hexadecimal
SPI	0A 00
KIC	00
KID	11
TAR	00 00 00
CNTR	00 00 00 00 01
PCNTR	00
Key	01 23 45 67 89 AB CD EF

4.7.3.2 Loading components

Cap files in loading scripts shall not include the descriptor component as described in Java Card 2.1 VM Architecture Specification [13].

4.8 Testing methodology

4.8.1 Test interfaces and facilities

The SIM-ME interface provides the main transport interface for the purpose of performing conformance tests.

The SIM API interface provides the main test interface for the purpose of performing conformance tests.

5 Test plan

The test plan is divided according to the SIM API specification, that way the tests will follow the class hierarchy for the sim.toolkit and sim.access package; for the SIM Toolkit framework this test plan describes the different points that will be tested with the present test specification.

6 API Test Plan

6.1 Package sim.access:

6.1.1 Interface SIMView

NOTE: The Test applet shall be run on a class that implements this interface.

6.1.1.1 Constants

Test Area Reference: API_1_SVW_CONST

6.1.1.1.1 Conformance Requirements

This subclause does not describe the conformance requirements for a method, but rather for the constants of the interface.

6.1.1.1.1 Normal execution

CRRN1: The constants shall have the same name and value that is defined in 3GPP TS 43.019 [7].

6.1.1.1.2 Test Suite Files

None.

6.1.1.1.3 Test Procedure

The constants in Java are resolved at compilation time, therefore a runtime test is not useful. No test of constants will be performed.

6.1.1.2 Method select(short fid, byte[] fci, short fciOffset, short fciLength)

Test Area Reference: API 1 SVW SLCTS BSS

6.1.1.2.1 Conformance Requirements

The method with the following header shall be compliant to its definition in the API.

6.1.1.2.1.1 Normal execution

- CRRN1: If the desired file is selected, the length of the FCI (File Control Information) which has been written to the array fci is returned.
- CRRN2: If the length fciLength is greater than or equal to the length of the FCI structure, the whole FCI structure is copied into the array fci and the length of the FCI which has been written to the array fci is returned.
- CRRN3: If the length fciLength is less than the length of the FCI structure, the first part of the FCI structure is copied into the array fci and the length of the FCI which has been written to the array fci is returned.
- CRRN4: After selecting a DF/MF no EF is selected.
- CRRN5: After selecting a linear fixed EF no record is selected.

- CRRN6: After selecting a cyclic EF the first record which is the last updated record is selected.
- CRRN7: The current files (file context) of any other applets shall not be changed. See TS 43.019 [7] §5.2. This will be tested during the testing of the framework.
- CRRN8: The information returned by fci shall be formatted as described in TS 51.011 [3], §9.2.1.
- CRRN9: The file with a File-ID that matches fid shall be found according to the following selection rules:
 - 1) An immediate child EF or DF of the current MF/DF can be selected,
 - 2) A sibling DF of the current DF can be selected,
 - 3) The current MF/DF it self can be selected,
 - 4) The parent MF/DF of the current DF can be selected,
 - 5) The MF can always be selected.

6.1.1.2.1.2 Parameter errors

- CRRP1: If the array fci is null, an instance of NullPointerException shall be thrown.
- CRRP2: If fciOffset is less than 0, an instance of ArrayIndexOutOfBoundsException shall be thrown.
- CRRP3: If fciLength is less than 0, an instance of ArrayIndexOutOfBoundsException shall be thrown.
- CRRP4: If fciOffset plus fciLength is greater than the length of the array fci.length, or fciOffset equals fci.length, an instance of ArrayIndexOutOfBoundsException shall be thrown.

6.1.1.2.1.3 Context errors

- CRRC1: If the file with a File-ID which matches fid could not be found according to the selection rules listed in CRRN9, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.FILE_NOT_FOUND.
- CRRC2: If the method call causes a memory problem (e.g. memory access error), an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.MEMORY_PROBLEM.
- CRRC3: If the method call causes an error to occur that is not expected and thus not handled, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.INTERNAL_ERROR.

6.1.1.2.2 Test Suite Files

Additional requirements for the GSM personalization: None

Test Script: API_1_SVW_SLCTS_BSS_1.scr

Test Applet: API_1_SVW_SLCTS_BSS_1.java

Load Script: API_1_SVW_SLCTS_BSS_1.ldr

Cleanup Script: API_1_SVW_SLCTS_BSS_1.clr

Parameter File: API_1_SVW_SLCTS_BSS_1.par

6.1.1.2.3 Test Procedure

ld	Description	API Expectation	APDU Expectation
0	SIM Initialization	Responses ignored.	
1	Select EFICCID in MF (Transparent EF)	No exception shall be thrown.	
	fid = SIMView.FID_EF_ICCID	Shall return a value not greater	
	<pre>byte[] fci = new byte[34]</pre>	than 20.	
	<pre>fciOffset = 0 fciLength = 20</pre>		
	select()	<description fci:<="" of="" th=""><th></th></description>	
		XX XX	
		XX XX	
		2F E2	
		04	
2	Select EF _{ICCID} in MF (Transparent EF)	No exception shall be thrown.	
-	fid = SIMView.FID_EF_ICCID	Shall return 13.	
	fciOffset = 0	fci shall contain the first 13 bytes of	
	fciLength = 13	the FCI structure.	
	select()		
3	Select DF _{GSM} in MF	No exception shall be thrown.	
	<pre>fid = SIMView.FID_DF_GSM fciOffset = 0</pre>	Shall return 7.	
	fciLength = 7	fci shall contain the first 7 bytes	
	select()	of the FCI.	
		<description fci:<="" of="" th=""><th></th></description>	
		XX XX XX XX	
		7F 20	
		02	
		>	
3	Select DF _{GSM} in MF	No exception shall be thrown.	
	fid = SIMView.FID_DF_GSM	Shall return 7.	
	fciOffset = 0	fci shall contain the entire FCI	
	fciLength = 7	structure.	
	select()	<description fci:<="" of="" th=""><th></th></description>	
		XX XX	
		XX XX	
		7F 20	
		02	
L.	0-1(FF :: DF (0::-1:-FF)	>	
4	Select EF _{ACM} in DF _{GSM} (CyclicEF) fid = SIMView.FID_EF_ACM	No exception shall be thrown.	
	fciOffset = 0	Shall return a value between 15 and 20. (Cyclic EF)	
	fciLength = 20	fci shall contain the first 15 or more	
	select()	bytes of the FCI structure.	
		fci[14] shall have the value 3	
		(length of record).	
5	Select MF	No exception shall be thrown.	
	fid = SIMView.FID_MF	Shall return a value between 22	
	fciOffset = 0	and 34.	
	<pre>fciLength = 34 select()</pre>	fci shall contain the entire FCI	
	, ,	structure.	
6	Select DF _{TELECOM} in MF	No exception shall be thrown.	
	<pre>fid = SIMView.FID_DF_TELECOM fci[0] = fci[1] = '05'</pre>	Shall return 20.	
	fciOffset = 2	fci shall contain the first 20 bytes of	
	fciLength = 20	the FCI structure starting at index	
	select()	2. The first two bytes shall (still)	
7	Select EF _{FDN} in DF _{TELECOM} (Linear FixedEF)	have the value '05'. No exception shall be thrown.	
'	fid = SIMView.FID_EF_FDN	Shall return 15.	
	fciOffset = 0	fci shall contain the first 15 bytes of	
	fciLength = 15	the FCI structure.	
	select()	fci[14] shall have the value 28	
		(length of record).	
8	fci is null	Shall throw	
1	fid = SIMView.FID_EF_FDN	java.lang.NullPointerException.	
	<pre>byte[] nullBuffer = null</pre>		
	fciOffset = 0		
	<pre>fciLength = 15 select()</pre>		
Ь	202001/		

ld	Description	API Expectation	APDU Expectation
9	fciOffset < 0	Shall throw	-
	fid = SIMView.FID_EF_FDN	java.lang.ArrayIndexOutOfBoundsE	
	fciOffset = -1 fciLength = 15	xception.	
	select()		
10	fciLength < 0	Shall throw	
	fid = SIMView.FID_EF_FDN	java.lang.ArrayIndexOutOfBoundsE	
	fciOffset = 0	xception.	
	<pre>fciLength = -1 select()</pre>		
11	fciOffset + fciLength > fci.length	Shall throw	
11	fid = SIMView.FID_EF_FDN	java.lang.ArrayIndexOutOfBoundsE	
	fciOffset = 20	xception.	
	fciLength = 15	Noop no	
40	select()	01 11 11	
12	fciOffset fci.length fid = SIMView.FID_EF_FDN	Shall throw	
	fciOffset = 34	java.lang.ArrayIndexOutOfBoundsE	
	fciLength = 1	xception	
	select()		
13	Selection possibilities	1 - No exception shall be thrown.	
	<pre>1 - fid = SIMView.FID_MF fciOffset = 0</pre>	2 - No exception shall be thrown.	
	fciLength = 15	3 - No exception shall be thrown.	
	select()	4 - No exception shall be thrown.	
	2 - fid = SIMView.FID_DF_TELECOM	5 - No exception shall be thrown.	
	select()	6 - No exception shall be thrown.	
	<pre>3 - fid = SIMView.FID_DF_GRAPHICS select()</pre>	7 - No exception shall be thrown.8 - No exception shall be thrown.	
	4 - fid = SIMView.FID_DF_TELECOM	9 - No exception shall be thrown.	
	select()	o No exception shall be thrown.	
	5 - fid = SIMView.FID_DF_GRAPHICS		
	select()		
	<pre>6 - fid = SIMView.FID_MF select()</pre>		
	7 - fid = SIMView.FID_DF_GSM		
	select()		
	8 - fid = SIMView.FID_DF_TELECOM		
	<pre>select() 9 - fid = SIMView.FID_DF_TELECOM</pre>		
	select()		
	_		
14	EF not selected after MF/DF selection	1 - No exception shall be thrown.	
1	<pre>1 - fid = SIMView.FID_MF select()</pre>	2 - Shall throw	
1	fid = SIMView.FID_EF_ICCID	sim.access.SIMViewException with	
	select()	reason code NO_EF_SELECTED.	
1	2 - fid = SIMView.FID_MF		
	select()		
15	readBinary() No selection of non-reachable file	1 - No exception shall be thrown.	
15	1 - fid = SIMView.FID_MF	2 - Shall throw	
	select()	sim.access.SIMViewException with	
	2 - fid = SIMView.FID_EF_ACM	reason code FILE_NOT_FOUND.	
40	select()		
16	No record is selected after selecting linear	1 - No exception shall be thrown.	
	fixed EF 1 - fid = SIMView.FID_MF	2 - No exception shall be thrown.3 - No exception shall be thrown.	
	select()	4 - Shall throw	
1	2 - fid = FID_DF_SIMTEST	sim.access.SIMViewException with	
	select()	reason code	
	3 - fid = FID_EF_LARU	RECORD_NUMBER_NOT_AVAIL	
	<pre>select() 4 - recNumber = 0</pre>	ABLE.	
	mode = REC_ACC_MODE_ABSOLUTE_CURRENT		
	readRecord()		

ld	Description	API Expectation	APDU Expectation
17	Record pointer in selected cyclic EF 1 - fid = SIMView.FID_MF select() 2 - fid = FID_DF_SIMTEST select() 3 - fid = FID_EF_CARU select() 4 - byte[] data1 = { 1,2,3 } mode = REC_ACC_MODE_PREVIOUS updateRecord(data1) 5 - fid = FID_EF_CARU select() readRecord(data2) compare data1 to data2	No exception shall be thrown. The contents of data1 and data2 shall be identical.	·

23

6.1.1.2.4 Test Coverage

CRR Number	Test Case Number	
N1	1-7	
N2	3, 5	
N3	1, 2, 4, 6, 7	
N4	14	
N5	16	
N6	17	
N8	1, 3	
N9	1-7, 13	
P1	8	
P2	9	
P3	10	
P4	11, 12	
C1	15	
C2, C3	Not Tested	

6.1.1.3 Method select (short fid)

Test Area Reference: API_1_SVW_SLCTS

6.1.1.3.1 Conformance Requirements

The method with the following header shall be compliant to its definition in the API.

6.1.1.3.1.1 Normal execution

- CRRN1: If the desired file is selected, no exception is thrown.
- CRRN2: After selecting a DF/MF no EF is selected.
- CRRN3: After selecting a linear fixed EF no record is selected.
- CRRN4: After selecting a cyclic EF the first record which is the last updated record is selected.
- CRRN5: The current files (file context) of any other applets shall not be changed [TS 43.019 [7] §5.2]. This will be tested during the testing of the framework.
- CRRN6: The file with a File-ID that matches fid shall be found according to the following selection rules:
 - 1) An immediate child EF or DF of the current MF/DF can be selected,
 - 2) A sibling DF of the current DF can be selected,
 - 3) The current MF/DF it self can be selected,

- 4) The parent MF/DF of the current DF can be selected,
- 5) The MF can always be selected.

6.1.1.3.1.2 Parameter errors

No requirements.

6.1.1.3.1.3 Context errors

- CRRC1: If the file with a File-ID which matches fid could not be found according to the selection rules listed in CCRN6, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.FILE NOT FOUND.
- CRRC2: If the method call causes a memory problem (e.g. memory access error), an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.MEMORY_PROBLEM.
- CRRC3: If the method call causes an error to occur that is not expected and thus not handled, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.INTERNAL_ERROR.

6.1.1.3.2 Test Suite Files

Additional requirements for the GSM personalization: None

Test Script: API_1_SVW_SLCTS_1.scr

Test Applet: API_1_SVW_SLCTS_1.java

Load Script: API_1_SVW_SLCTS_1.ldr

Cleanup Script: API_1_SVW_SLCTS_1.clr

Parameter File: API_1_SVW_SLCTS_1.par

6.1.1.3.3 Test Procedure

ld	Description	API Expectation	APDU Expectation
0	SIM Initialization	Responses ignored.	
1	Select EF _{ICCID} in MF (Transparent EF) fid = SIMView.FID_EF_ICCID select()	No exception shall be thrown.	
2	<pre>EF not selected after MF/DF selection 1 - fid = SIMView.FID_MF select() fid = SIMView.FID_EF_ICCID select() 2 - fid = SIMView.FID_MF select() readBinary()</pre>	No exception shall be thrown. Shall throw sim.access.SIMViewException with reason code NO_EF_SELECTED.	
3	No record is selected after selecting linear fixed EF 1 - fid = SIMView.FID_MF select() 2 - fid = FID_DF_SIMTEST select() 3 - fid =FID_EF_LARU select() 4 - recNumber = 0 mode = REC_ACC_MODE_ABSOLUTE_CURRENT readRecord()	 1 - No exception shall be thrown. 2 - No exception shall be thrown. 3 - No exception shall be thrown. 4 - Shall throw sim.access.SIMViewException with reason code RECORD_NUMBER_NOT_AVAIL ABLE. 	

ld	Description	API Expectation	APDU Expectation
4	Record pointer in selected cyclic EF	1 - No exception shall be thrown.	
	<pre>1 - fid = SIMView.FID_MF select()</pre>	2 - No exception shall be thrown.	
	2 - fid =FID_DF_SIMTEST	3 - No exception shall be thrown.	
	select()	4 - No exception shall be thrown.	
	3 - fid = FID_EF_CARU	5 - The contents of data1 and data2	
	select()	shall be identical.	
	4 - byte[] data1 = { 1,2,3 }		
	updateRecord(data1)		
	5 - fid = FID_EF_CARU		
	<pre>select() readRecord(data2)</pre>		
	compare data1 to data2		
5	Selection possibilities	1 - No exception shall be thrown.	
3	1 - fid = SIMView.FID_MF	2 - No exception shall be thrown.	
	select()	3 - No exception shall be thrown.	
	2 - fid = SIMView.FID_DF_TELECOM	4 - No exception shall be thrown.	
	select()	5 - No exception shall be thrown.	
	3 - fid = SIMView.FID_DF_GRAPHICS	6 - No exception shall be thrown.	
	select()	7 - No exception shall be thrown.	
	<pre>4 - fid = SIMView.FID_DF_TELECOM select()</pre>	8 - No exception shall be thrown.	
	5 - fid = SIMView.FID_DF_GRAPHICS	9 - No exception shall be thrown.	
		9 - No exception shall be thrown.	
	6 - fid = SIMView.FID_MF		
	select()		
	7 - fid = SIMView.FID_DF_GSM		
	select()		
	8 - fid = SIMView.FID_DF_TELECOM		
	select()		
	9 - fid = SIMView.FID_DF_TELECOM select()		
6	No selection of unreachable file	1 - No exception shall be thrown.	
	1 - fid = SIMView.FID_MF	2 - Shall throw	
	select()	sim.access.SIMViewException with	
	2 - fid = SIMView.FID_EF_ACM	•	
	select()	reason code FILE_NOT_FOUND.	

6.1.1.3.4 Test Coverage

CRR Number	Test Case Number	
N1	1	
N2	2	
N3	3	
N4	4	
N6	5	
C1	6	
C2, C3	Not Tested	

6.1.1.4 Method status

Test Area Reference: API_1_SVW_STAT_BSS

6.1.1.4.1 Conformance Requirements

The method with the following header shall be compliant to its definition in the API.

6.1.1.4.1.1 Normal execution

• CRRN1: The FCI (File Control Information) of the current DF (or MF) is returned in the same format as for a SELECT command in case of selecting an MF/DF (described in 3GPP TS 43.019 [7], subclause 9.2.1).

- CRRN2: If the length fciLength is greater than or equal to the length of the FCI structure, the whole FCI structure is copied into the array fci and the length of the FCI which has been written to the array fci is returned.
- CRRN3: If the length fciLength is less than the length of the FCI structure, the first part of the FCI structure is copied into the array fci and the length of the FCI which has been written to the array fci is returned.

6.1.1.4.1.2 Parameter errors

- CRRP1: If the array fci is null, an instance of NullPointerException shall be thrown.
- CRRP2: If fciOffset is less than 0, an instance of ArrayIndexOutOfBoundsException shall be thrown.
- CRRP3: If fciLength is less than 0, an instance of ArrayIndexOutOfBoundsException shall be thrown.
- CRRP4: If fciOffset plus fciLength is greater than the length of the array fci.length, or fciOffset equals fci.length, an instance of ArrayIndexOutOfBoundsException shall be thrown.

6.1.1.4.1.3 Context errors

- CRRC1: If the method call causes a memory problem (e.g. memory access error), an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.MEMORY PROBLEM.
- CRRC2: If the method call causes an error to occur that is not expected and thus not handled, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.INTERNAL_ERROR.

6.1.1.4.2 Test Suite Files

Additional requirements for the GSM personalization:

Test Script: API_1_SVW_STAT_BSS_1.scr

Test Applet: API_1_SVW_STAT_BSS_1.java

Load Script: API_1_SVW_STAT_BSS_1.ldr

Cleanup Script: API_1_SVW_STAT_BSS_1.clr

Parameter File: API_1_SVW_STAT_BSS_1.par

6.1.1.4.3 Test Procedure

ld	Description	API Expectation	APDU Expectation
0	SIM Initialization	Responses ignored.	
1	Status of MF byte[] fci = new byte[34] fciOffset = 0 fciLength = 7 status()	No exception shall be thrown. Shall return 7. fci shall contain the entire FCI structure. <description 00="" 01="" 3f="" fci:="" of="" xx=""></description>	
2	Status after select EF _{ICCID} in MF 1 - fid = SIMView.FID_DF_GSM fciOffset = 0 fciLength = 34 len = select() 2 - byte[] fci2 = new byte[34] len2 = status() 3 - Compare len and len2 4 - Compare the len bytes of fci and fci2	1 - No exception shall be thrown. Shall return a value between 22 and 34. 2 - No exception shall be thrown. Shall return 22 or more. 3 - Ien and Ien2 shall be identical 4 - fci and fci2 shall be identical	

ld	Description	API Expectation	APDU Expectation
3	Status of DF _{Telecom} 1 - fid = SIMView.FID_DF_TELECOM select() 2 - fciOffset = 0 fciLength = 100 status()	1 - No exception shall be thrown. Shall return a value between 22 and 34. 2 - No exception shall be thrown. Shall return a value between 22 and34. fci shall contain the entire FCI structure (check that returned value is equal to 13 plus the "length of following data" - fci[12]).FID of the returned fci (fci[4:5]) is FID_DF_TELECOM.	
4	Status DF _{TELECOM} fciOffset = 0 fciLength = 7 status()	No exception shall be thrown. Shall return 7. fci shall contain the first 7 bytes of the FCI structure starting at index 0. FID of the returned fci (fci[4:5]) is FID DF TELECOM.	
5	<pre>fci is null byte[] nullBuffer = null fciOffset = 0 fciLength = 34 status()</pre>	Shall throw java.lang.NullPointerException.	
6	<pre>fciOffset < 0 fciOffset = -1 fciLength = 34 status()</pre>	Shall throw java.lang.ArrayIndexOutOfBoundsE xception.	
7	<pre>fciLength < 0 fciOffset = 0 fciLength = -1 status()</pre>	Shall throw java.lang.ArrayIndexOutOfBoundsE xception.	
8	<pre>fciOffset + fciLength > fci.length fciOffset = 20 fciLength = 15 status()</pre>	Shall throw java.lang.ArrayIndexOutOfBoundsE xception.	
9	<pre>fciOffset fci.length fciOffset = 34 fciLength = 1 status()</pre>	Shall throw java.lang.ArrayIndexOutOfBoundsE xception.	

6.1.1.4.4 Test Coverage

CRR Number	Test Case Number	
N1	1-4	
N2	2, 3	
N3	1, 4	
P1	5	
P2	6	
P3	7	
P4	8, 9	
C1, C2	Not Tested	

6.1.1.5 Method readBinary

Test Area Reference: API_1_SVW_REDBS_BSS

6.1.1.5.1 Conformance Requirements

The method with the following header shall be compliant to its definition in the API.

6.1.1.5.1.1 Normal execution

• CRRN1: If data can be accessed at the specified offset, the value respOffset plus respLength are returned and the data bytes of the currently selected transparent file are returned in resp.

6.1.1.5.1.2 Parameter errors

- CRRP1: If fileOffset is less than 0, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.OUT_OF_FILE_BOUNDARIES.
- CRRP2: If fileOffset plus respLength exceeds the length of the file, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.OUT_OF_FILE_BOUNDARIES.
- CRRP3: If the array resp is null, an instance of NullPointerException shall be thrown.
- CRRP4: If respOffset is less than 0, an instance of ArrayIndexOutOfBoundsException shall be thrown.
- CRRP5: If respLength is less than 0, an instance of ArrayIndexOutOfBoundsException shall be thrown.
- CRRP6: If respOffset plus respLength is greater than the length of the array resp.length, an instance of ArrayIndexOutOfBoundsException shall be thrown.

6.1.1.5.1.3 Context errors

- CRRC1: If the calling applet has currently no EF selected, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.NO_EF_SELECTED.
- CRRC2: If the currently selected EF is not transparent, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.FILE_INCONSISTENT.
- CRRC3: If the calling applet does not fulfil the access condition, READ, to perform this function, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.AC_NOT_FULFILLED.
- CRRC4: If the currently selected EF is invalidated and the file status of the EF does not allow for the reading of an invalidated file, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.INVALIDATION_STATUS_CONTRADICTION.
- CRRC5: If the method call causes a memory problem (e.g. memory access error), an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.MEMORY_PROBLEM.
- CRRC6: If the method call causes an error to occur that is not expected and thus not handled, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.INTERNAL_ERROR.

6.1.1.5.2 Test Suite Files

Additional requirements for the GSM personalization: none.

Test Script: API_1_SVW_REDBS_BSS_1.scr
Test Applet: API_1_SVW_REDBS_BSS_1.java
Load Script: API_1_SVW_REDBS_BSS_1.ldr

Cleanup Script: API_1_SVW_REDBS_BSS_1.clr
Parameter File: API_1_SVW_REDBS_BSS_1.par

6.1.1.5.3 Test Procedure

ld	Description	API Expectation	APDU Expectation
0	SIM Initialization	Responses ignored	
1	<pre>Read from EFICCID in MF (Transparent EF) 1 - fid = SIMView.FID_EF_ICCID select() 2 - fileOffset = 0 byte[] resp = new byte[20] resp[0:19] = '55' respOffset = 10</pre>	1 - No exception shall be thrown. 2 - No exception shall be thrown. Shall return 20. resp shall contain the entire contents of EFICCID starting at index 10.	
	respLength = 10 readBinary()	<pre><description 0f="" 55="" ff="" of="" resp:=""></description></pre>	
2	Read from EFICCID in MF resp[0:19] = '55' fileOffset = 5 respOffset = 10 respLength = 5 readBinary()	No exception shall be thrown. Shall return 15. resp shall contain the last 5 bytes of EFICCID starting at index 10. <description 55="" ff="" of="" resp:=""></description>	
3	Offset into File out of bounds fileOffset = -1 respOffset = 0 respLength = 10 readBinary()	Shall throw sim.access.SIMViewException with reason code OUT_OF_FILE_BOUNDARIES.	
4	<pre>fileOffset + respLength > EF length fileOffset = 9 respOffset = 0 respLength = 2 readBinary()</pre>	Shall throw sim.access.SIMViewException with reason code OUT_OF_FILE_BOUNDARIES.	
5	<pre>resp is null byte[] nullBuffer = null fileOffset = 0 respOffset = 0 respLength = 10 readBinary()</pre>	Shall throw java.lang.NullPointerException.	
6	<pre>respOffset < 0 fileOffset = 0 respOffset = -1 respLength = 10 readBinary()</pre>	Shall throw java.lang. ArrayIndexOutOfBoundsException.	
7	<pre>respLength < 0 fileOffset = 0 respOffset = 0 respLength = -1 readBinary()</pre>	Shall throw java.lang. ArrayIndexOutOfBoundsException.	
8	<pre>respOffset + respLength > resp.length fileOffset = 0 respOffset = 10 respLength = 11 readBinary()</pre>	Shall throw java.lang. ArrayIndexOutOfBoundsException.	
9	<pre>EF is not Transparent 1 - fid = FID_DF_SIMTEST select() 2 - fid = FID_EF_LARU select() 3 - fileOffset = 0 respOffset = 0 respLength = 1 readBinary()</pre>	1 - No exception shall be thrown.2 - No exception shall be thrown.3 - Shall throw sim.access.SIMViewException with reason code FILE_INCONSISTENT.	

ld	Description	API Expectation	APDU Expectation
10	Access condition not fulfilled 1 - fid = DFSIMTTEST select() 2 - fid = EFTNR select() 3 - fileOffset = 0 respOffset = 0 respLength = 1 readBinary()	Shall throw sim.access.SIMViewException with reason code AC_NOT_FULFILLED.	·
11	EF is invalidated 1 - fid = EFTNU invalidate() 2 - readBinary() 3 - rehabilitate()	1 - No exception shall be thrown. 2 - Shall throw sim.access.SIMViewException with reason code INVALIDATION_STATUS_CONTR ADICTION. 3 - No exception shall be thrown.	
12	No EF selected 1- fid = SIMView.FID_MF select() 2 readBinary()	1 - No exception shall be thrown. 2 - Shall throw sim.access.SIMViewException with reason code NO_EF_SELECTED.	

6.1.1.5.4 Test Coverage

CRR Number	Test Case Number
N1	1-2
P1	3
P2	4
P3	5
P4	6
P5	7
P6	8,
C1	12
C2	9
C3	10
C4	11
C5, C6	Not Tested

6.1.1.6 Method updateBinary

Test Area Reference: API_1_SVW_UPDBS_BSS

6.1.1.6.1 Conformance Requirements

The method with the following header shall be compliant to its definition in the API.

6.1.1.6.1.1 Normal execution

• CRRN1: The currently selected transparent file is updated starting at fileOffset, with the string of dataLength bytes in the array data starting at dataOffset.

6.1.1.6.1.2 Parameter errors

• CRRP1: If fileOffset is less than 0, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.OUT_OF_FILE_BOUNDARIES.

- CRRP2: If fileOffset plus dataLength exceeds the length of the file, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.OUT_OF_FILE_BOUNDARIES.
- CRRP3: If the array data is null, an instance of NullPointerException shall be thrown.
- CRRP4: If dataOffset is less than 0, an instance of ArrayIndexOutOfBoundsException shall be thrown.
- CRRP5: If dataLength is less than 0, an instance of ArrayIndexOutOfBoundsException shall be thrown.
- CRRP6: If dataOffset plus dataLength greater than the length of the array data.length an instance of ArrayIndexOutOfBoundsException shall be thrown.

6.1.1.6.1.3 Context errors

- CRRC1: If the calling applet has currently no EF selected, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.NO_EF_SELECTED.
- CRRC2: If the currently selected EF is not transparent, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.FILE_INCONSISTENT.
- CRRC3: If the calling applet does not fulfil the access condition, UPDATE, to perform this function, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.AC_NOT_FULFILLED.
- CRRC4: If the currently selected EF is invalidated and the file status of the EF does not allow for updating of an invalidated file, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.INVALIDATION_STATUS_CONTRADICTION.
- CRRC5: If the method call causes a memory problem (e.g. memory access error), an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.MEMORY_PROBLEM.
- CRRC6: If the method call causes an error to occur that is not expected and thus not handled, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.INTERNAL_ERROR.

6.1.1.6.2 Test Suite Files

Additional requirements for the GSM personalization: None

Test Script: API_1_SVW_UPDBS_BSS_1.scr

Test Applet: API_1_SVW_UPDBS_BSS_1.java

Load Script: API_1_SVW_UPDBS_BSS_1.ldr

Cleanup Script: API_1_SVW_UPDBS_BSS_1.clr

Parameter File: API_1_SVW_UPDBS_BSS_1.par

6.1.1.6.3 Test Procedure

ld	Description	API Expectation	APDU Expectation
0	SIM Initialization	Responses ignored.	
1	No EF selected	Shall throw	
		sim.access.SIMViewException with	
	byte[] data = new byte[20]	reason code NO_EF_SELECTED.	
	data[0] = '55'		
	dataOffset = 0		
	dataLength = 10		
	updateBinary()		

ld	Description	API Expectation	APDU Expectation
2	Update Transparent EF	1 - No exception shall be thrown.	•
	1 - fid = DFSIMTEST	2 - No exception shall be thrown.	
	select()	3 - No exception shall be thrown.	
	2 - fid = EFTARU	4 - No exception shall be thrown.	
	select() 3 - fileOffset = 3	Data in resp[0] shall be '55'.	
	data[0] = '55'	1.1.1	
	dataOffset = 0		
	dataLength = 1		
	updateBinary()		
	4 - fileOffset = 3		
	respOffset = 0		
	respLength = 1		
3	readBinary() 1 - fileOffset = 254	1 - No exception shall be thrown.	
3	data[0] = '55'	2 - No exception shall be thrown.	
	data[1] = 'AA'	Data in resp shall be	
	data[2] = '66'	resp[0] = '55'	
	dataOffset = 0	resp[0] = 33 resp[1] = 'AA'	
	dataLength = 3	resp[1] = AA resp[2] = '66'	
	updateBinary() 2 - fileOffset = 254	165ρ[2] - 00	
	2 - IlleOIIset = 254 respOffset = 0		
	respLength = 3		
	readBinary()		
4	Offset into File out of bounds	Shall throw	
	fileOffset = -1	sim.access.SIMViewException with	
	dataOffset = 0	reason code	
	dataLength = 10	OUT_OF_FILE_BOUNDARIES.	
5	updateBinary()	Chall throw	
Э	fileOffset + dataLength > EF length	Shall throw	
	dataOffset = 0	sim.access.SIMViewException with reason code	
	dataLength = 2	OUT_OF_FILE_BOUNDARIES.	
	updateBinary()	OUT_OF_FILE_BOUNDARIES.	
6	data is null	Shall throw	
	byte[] nullBuffer = null	java.lang.NullPointerException.	
	fileOffset = 0		
	<pre>dataOffset = 0 dataLength = 10</pre>		
	updateBinary()		
7	dataOffset < 0	Shall throw	
	fileOffset = 0	java.lang.	
		ArrayIndexOutOfBoundsException.	
	dataOffset = -1		
	dataLength = 10		
8	updateBinary() dataLength < 0	Shall throw	
"	fileOffset = 0	java.lang.	
	dataOffset = 0	ArrayIndexOutOfBoundsException.	
	dataLength = -1	ay.iiaa.iaataibaailaaExoopiioii.	
	updateBinary()		
9	dataOffset + dataLength > data.length	Shall throw	
	<pre>fileOffset = 0 dataOffset = 10</pre>	java.lang.	
	dataOffset = 10 dataLength = 11	ArrayIndexOutOfBoundsException.	
	updateBinary()		
10	EF is not Transparent	1 - No exception shall be thrown.	
	1 - fid = FID_DF_SIMTEST	2 - No exception shall be thrown.	
	select()	3 - Shall throw	
	2 - fid = FID_EF_LARU	sim.access.SIMViewException with	
	select()	reason code	
	<pre>3 - fileOffset = 0 data[0] = '55'</pre>	FILE_INCONSISTENT.	
	dataOffset = 0		
	dataLength = 1		
	updateBinary()		
	-	•	

ld	Description	API Expectation	APDU Expectation
11	Access condition not fulfilled 1 - fid = DFSIMTEST select() fid = EFTNU select() 2 - fileOffset = 0 data[0] = '55' dataOffset = 0 dataLength = 1 updateBinary()	1 - No exception shall be thrown. 2 - Shall throw sim.access.SIMViewException with reason code AC_NOT_FULFILLED.	•
12	<pre>EF is invalidated 1 - fid = EFTNR invalidate() 2 - fileOffset = 0 data[0] = '55' dataOffset = 0 dataLength = 1 updateBinary() 3 - rehabilitate()</pre>	No exception shall be thrown. Shall throw sim.access.SIMViewException with reason code INVALIDATION_STATUS_CONTR ADICTION. No exception shall be thrown.	

6.1.1.6.4 Test Coverage

CRR Number	Test Case Number
N1	2, 3
P1	4
P2	5
P3	6
P4	7
P5	8
P6	9
C1	1
C2	10
C3	11
C4	12
C5, C6	Not Tested

6.1.1.7 Method readRecord

Test Area Reference: API_1_SVW_REDRSBS_BSS

6.1.1.7.1 Conformance Requirements

The method with the following header shall be compliant to its definition in the API.

6.1.1.7.1.1 Normal execution

- CRRN1: The data bytes from the record, specified by mode and recNumber of the currently selected linear fixed or cyclic EF, is read at recOffset. A total of respLength bytes of this data is copied to the array resp at respOffset.
- CRRN2: If the access mode is REC_ACC_MODE_ABSOLUTE_CURRENT:
 - if recNumber is not 0, the record addressed by recNumber will be read;
 - if recNumber is 0 the current selected record will be read; and

- the current record pointer shall not change.
- CRRN3: If the access mode is REC ACC MODE NEXT:
 - the next record relative to the current selected record will be selected and read;
 - if no current record is selected, the first record will be selected and read;
 - if the current record pointer is set to the last record for a cyclic EF the record pointer is set to the first record and the record is read;
 - the current record pointer of any other applet shall not be changed.
- CRRN4: If the access mode is REC_ACC_MODE_PREVIOUS:
 - the previous record relative to the current selected record will be selected and read;
 - if no current record is selected, the last record will be selected and read;
 - if the current record pointer is set to the first record, for a linear fixed EF the method responses with an error exception and for a cyclic EF the record pointer is set to the last record and the record is read;
 - the current record pointer of any other applet shall not be changed.

6.1.1.7.1.2 Parameter errors

- CRRP1: If the currently selected EF is linear fixed and the access mode is REC_ACC_MODE_ABSOLUTE_CURRENT and recNumber is less than 0 or greater than records available, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.RECORD_NUMBER_NOT_AVAILABLE.
- CRRP2: If the currently selected EF is linear fixed and the access mode is REC_ACC_MODE_ABSOLUTE_CURRENT, recNumber is 0 and there is no current record selected, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.RECORD_NUMBER_NOT_AVAILABLE.
- CRRP3: If the currently selected EF is linear fixed and the access mode is REC_ACC_MODE_NEXT and the current record pointer is set to the last record, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.RECORD_NUMBER_NOT_AVAILABLE.
- CRRP4: If the currently selected EF is linear fixed and the access mode is REC_ACC_MODE_PREVIOUS and the current record pointer is set to the first record, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.RECORD_NUMBER_NOT_AVAILABLE.
- CRRP5: If the specified offset into the selected record recOffset is less than 0, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.OUT OF RECORD BOUNDARIES.
- CRRP6: If recOffset plus respLength is greater than the record length, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.OUT OF RECORD BOUNDARIES.
- CRRP7: If the access mode is not between 2 and 4 inclusive (2 = REC_ACC_MODE_NEXT, etc.), an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.INVALID_MODE.
- CRRP8: If the array resp is null, an instance of NullPointerException shall be thrown.
- CRRP9: If respOffset is less than 0, an instance of ArrayIndexOutOfBoundsException shall be thrown.
- CRRP10: If respLength is less than 0, an instance of ArrayIndexOutOfBoundsException shall be thrown.
- CRRP11: If respOffset plus respLength is greater than the length of the array resp.length, or respOffset equals resp.length, an instance of ArrayIndexOutOfBoundsException shall be thrown.

6.1.1.7.1.3 Context errors

- CRRC1: If the calling applet has currently no EF selected, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.NO EF SELECTED.
- CRRC2: If the currently selected EF is neither linear fixed nor cyclic, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.FILE INCONSISTENT.
- CRRC3: If the calling applet does not fulfil the access condition, READ, to perform this function, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.AC_NOT_FULFILLED.
- CRRC4: If the currently selected EF is invalidated and the file status of the EF does not allow for reading an invalidated file, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.INVALIDATION_STATUS_CONTRADICTION.
- CRRC5: If the method call causes a memory problem (e.g. memory access error), an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.MEMORY PROBLEM.
- CRRC6: If the method call causes an error to occur that is not expected and thus not handled, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.INTERNAL_ERROR.

6.1.1.7.2 Test Suite Files

Additional requirements for the GSM personalization: None

Test Script: API_1_SVW_REDRSBS_BSS_1.scr

Test Applet: API_1_SVW_REDRSBS_BSS_1.java

Load Script: API_1_SVW_REDRSBS_BSS_1.ldr

Cleanup Script: API_1_SVW_REDRSBS_BSS_1.clr

Parameter File: API_1_SVW_REDRSBS_BSS_1.par

6.1.1.7.3 Test Procedure

ld	Description	API Expectation	APDU Expectation
0	SIM Initialization	Responses ignored.	
1	No EF selected	Shall throw	
	recNumber = 1	sim.access.SIMViewException with	
	mode = REC_ACC_MODE_ABSOLUTE_CURRENT	reason code NO_EF_SELECTED.	
	recOffset = 0		
	<pre>byte[] resp = new byte[20]</pre>		
	respOffset = 0		
	respLength = 10		
	readRecord()		

ld	Description	API Expectation	APDU Expectation
2	Read Absolute and Current from Linear Fixed	1 - No exception shall be thrown.	-
	EF	2 - No exception shall be thrown.	
	1 - fid = DFSIMTEST	3 - No exception shall be thrown.	
	select()	resp shall be:	
	2 - fid = EFLARU	resp[0] = '55'	
	select()	1	
	// Record pointer not set.	resp[1] = '55'	
	3 - recNumber = 0	resp[2] = '55'	
	mode = REC_ACC_MODE_NEXT	resp[3] = '55'	
	recOffset = 0	4 - No exception shall be thrown.	
	respOffset = 0	resp shall be:	
	respLength = 4	resp[0] = 'AA'	
	readRecord() 4 - recNumber = 2	resp[1] = 'AA'	
	mode = REC_ACC_MODE_ABSOLUTE_CURRENT	resp[2] = 'AA'	
	readRecord()	resp[3] = 'AA'	
	5 - recNumber = 1	5 - No exception shall be thrown.	
	readRecord()	resp shall be:	
	6 - recNumber = 0	resp[0] = '55'	
	resp[0] = resp[1] = resp[2] = resp[3] =	resp[1] = '55'	
	'00'	resp[2] = '55'	
	readRecord()		
		resp[3] = '55'	
		6 - No exception shall be thrown.	
		resp shall be:	
		resp[0] = '55'	
		resp[1] = '55'	
		resp[2] = '55'	
		resp[3] = '55'	
3	Read Next from Linear Fixed EF	No exception shall be thrown.	
	recNumber = 0	resp shall be:	
	mode = REC_ACC_MODE_NEXT	resp[0] = 'AA'	
	recOffset = 0	resp[1] = 'AA'	
	respOffset = 0	resp[2] = 'AA'	
	respLength = 4	resp[3] = 'AA'	
	readRecord()		
4	Read Next from Linear Fixed EF	Shall throw	
	recNumber = 0	sim.access.SIMViewException with	
	mode = REC_ACC_MODE_NEXT	reason code	
	recOffset = 0	RECORD_NUMBER_NOT_AVAIL	
	respOffset = 0	ABLE.	
	respLength = 4	ABLL.	
	readRecord()		
5	Read Previous from Linear Fixed EF	No exception shall be thrown.	
	recNumber = 0	resp shall be:	
	mode = REC_ACC_MODE_PREVIOUS	resp[0] = '55'	
	recOffset = 0 respOffset = 0	resp[1] = '55'	
	respLength = 4	resp[2] = '55'	
	readRecord()	resp[3] = '55'	
	I Cadalicoota()	1	
6	Read Previous from Linear Fixed EF	Shall throw	
•	recNumber = 0	sim.access.SIMViewException with	
	mode = REC_ACC_MODE_PREVIOUS	reason code	
	recOffset = 0	RECORD_NUMBER_NOT_AVAIL	
	respOffset = 0	ABLE.	
	respLength = 4	ADLE.	
	readRecord()		

ld	Description	API Expectation	APDU Expectation
7	Read Absolute and Current from Cyclic EF	1 - No exception shall be thrown.	-
	1 - fid = EFCARU	2 - No exception shall be thrown.	
	<pre>select() 2 - recNumber = 2</pre>	resp shall be:	
	mode = REC_ACC_MODE_ABSOLUTE_CURRENT	resp[0] = ' AA'	
	recOffset = 0	resp[1] = ' AA'	
	respOffset = 0	resp[2] = ' AA'	
	respLength = 3	3 - No exception shall be thrown.	
	readRecord()	resp shall be:	
	3 - recNumber = 1 readRecord()	resp[0] = '55'	
	4 - recNumber = 0	resp[1] = '55'	
	resp[0] = resp[1] = resp[2] = '00	resp[2] = '55' 4 - No exception shall be thrown.	
	readRecord()	resp shall be:	
		resp[0] = '55'	
		resp[0] = 35 resp[1] = '55'	
		resp[2] = '55'	
8	Read Next from Cyclic EF	No exception shall be thrown.	
	recNumber = 0	resp shall be:	
	mode = REC_ACC_MODE_NEXT	resp[0] = 'AA'	
	recOffset = 0	resp[1] = 'AA'	
	respOffset = 0 respLength = 3	resp[2] = 'AA'	
1	readRecord()		
9	Read Next from Cyclic EF	No exception shall be thrown.	
	recNumber = 0	resp shall be:	
	mode = REC_ACC_MODE_NEXT	resp[0] = '55'	
	recOffset = 0 respOffset = 0	resp[1] = '55'	
	respLength = 3	resp[2] = '55'	
	readRecord()		
10	Read Previous from Cyclic EF	No exception shall be thrown.	
	recNumber = 0	resp shall be:	
	<pre>mode = REC_ACC_MODE_PREVIOUS recOffset = 0</pre>	resp[0] = 'AA'	
	respOffset = 0	resp[1] = 'AA'	
	respLength = 3	resp[2] = 'AA'	
	readRecord()		
11	Read Previous from Cyclic EF recNumber = 0	No exception shall be thrown.	
	mode = REC_ACC_MODE_PREVIOUS	resp shall be:	
	recOffset = 0	resp[0] = '55'	
	respOffset = 0	resp[1] = '55'	
	respLength = 3	resp[2] = '55'	
40	readRecord()	4. No expension shall be through	
12	Read Absolute from Linear Fixed EF beyond	1 - No exception shall be thrown.2 - Shall throw	
	1 - fid = EFLARU	sim.access.SIMViewException with	
	select()	reason code	
	2 - recNumber = -1	RECORD_NUMBER_NOT_AVAIL	
	<pre>mode = REC_ACC_MODE_ABSOLUTE_CURRENT recOffset = 0</pre>	ABLE.	
	respOffset = 0	3 - Shall throw	
	respLength = 4	sim.access.SIMViewException with	
	readRecord()	reason code	
	<pre>3 - recNumber = 3 readRecord()</pre>	RECORD_NUMBER_NOT_AVAIL	
40	, ,	ABLE.	
13	No current record in linear fixed EF, read	1 - No exception shall be thrown.2 - Shall throw	
	1 - fid = EFLARU	sim.access.SIMViewException with	
	select() // No curr rec	reason code	
	2 - recNumber = 0 // curr rec	RECORD_NUMBER_NOT_AVAIL	
	mode = REC_ACC_MODE_ABSOLUTE_CURRENT	ABLE.	
	recOffset = 0 respOffset = 0	,	
	respLength = 4		
	readRecord()		

ld	Description	API Expectation	APDU Expectation
14	recOffset < 0	1 - No exception shall be thrown.	-
	1 - fid = EFLARU	2 - Shall throw	
	<pre>select() 2 - recNumber = 1 // rec 1</pre>	sim.access.SIMViewException with	
	mode = REC_ACC_MODE_ABSOLUTE_CURRENT	reason code OUT_OF_RECORD_BOUNDARIE	
	recOffset = -1	S.	
	respOffset = 0 respLength = 4	S.	
	readRecord()		
15	recOffset + respLength > Record Length	1 - No exception shall be thrown.	
	1 - fid = EFLARU select()	2 - Shall throw	
	2 - recNumber = 1	sim.access.SIMViewException with	
	mode = REC_ACC_MODE_ABSOLUTE_CURRENT	reason code OUT_OF_RECORD_BOUNDARIE	
	recOffset = 2	S.	
	respOffset = 0 respLength = 4		
	readRecord()		
16	Reading with invalid mode	1 - No exception shall be thrown.	
	1 - fid = EFLARU select()	2 - Shall throw	
	2 - recNumber = 0	sim.access.SIMViewException with	
	mode = 1	reason code INVALID_MODE. 3 - Shall throw	
	recOffset = 0 respOffset = 0	sim.access.SIMViewException with	
	respLength = 4	reason code INVALID_MODE.	
	readRecord()		
	3 - mode = 5		
17	readRecord() resp is null	Shall throw	
''	byte[] nullBuffer = null	java.lang.NullPointerException.	
	mode = REC_ACC_MODE_ABSOLUTE_CURRENT	,gg	
	respOffset = 0 respLength = 10		
	readRecord()		
18	respOffset < 0	Shall throw	
	respOffset = -1	java.lang.	
	respLength = 10	ArrayIndexOutOfBoundsException.	
	readRecord ()		
19	respLength < 0 respOffset = 0	Shall throw	
	respLength = -1	java.lang. ArrayIndexOutOfBoundsException.	
	readRecord ()	·	
20	respOffset + respLength > resp.length	Shall throw	
	respOffset = 10 respLength = 11	java.lang. ArrayIndexOutOfBoundsException.	
	readRecord ()	ArrayindexOutOrBoundsException.	
21	EF is neither Cyclic nor Linear Fixed	1 - No exception shall be thrown.	
	1 - fid = DFSIMTEST select()	2 - No exception shall be thrown.	
	2 - fid = EFTNU	3 - Shall throw sim.access.SIMViewException with	
	select()	reason code	
	3 - respOffset = 0 respLength = 4	FILE_INCONSISTENT.	
	readRecord()	_	
22	Access condition not fulfilled	1 - No exception shall be thrown.	
	1 - fid = EFCNR select()	2 - Shall throw	
	2 - respLength = 3	sim.access.SIMViewException with reason code	
1	readRecord()	AC_NOT_FULFILLED.	
23	EF is invalidated	1 - No exception shall be thrown.	
	1 - fid = EFCNU	2 - Shall throw	
	invalidate()	sim.access.SIMViewException with	
	<pre>2 - readRecord() 3 - rehabilitate()</pre>	reason code	
1		INVALIDATION_STATUS_CONTR	
		ADICTION.	
		3 - No exception shall be thrown.	

6.1.1.7.4 Test Coverage

CRR Number	Test Case Number	
N1	2-5, 7-11	
N2	2, 7	
N3	3, 8, 9	
N4	5, 10, 11	
P1	12	
P2	13	
P3	4	
P4	6	
P5	14	
P6	15	
P7	16	
P8	17	
P9	18	
P10	19	
P11	20	
C1	1	
C2	21	
C3	22	
C4	23	
C5, C6	Not Tested	

6.1.1.8 Method updateRecord

Test Area Reference: API_1_SVW_UPDRSBS_BSS

6.1.1.8.1 Conformance Requirements

The method with the following header shall be compliant to its definition in the API.

6.1.1.8.1.1 Normal execution

- CRRN1: dataLength bytes of the record specified by mode and recNumber of the current selected linear fixed or cyclic EF are updated at recOffset, by using the string of bytes in the array data starting at dataOffset.
- CRRN2: If the access mode is REC_ACC_MODE_ABSOLUTE_CURRENT and the file is a linear fixed EF:
 - the record addressed by recNumber will be updated;
 - if recNumber is 0 the current selected record will be updated; and
 - the current record pointer shall not change.
- CRRN3: If the access mode is REC_ACC_MODE_NEXT and the file is a linear fixed EF:
 - the next record relative to the current selected record will be selected and updated;
 - if no current record is selected, the first record will be selected and updated;
 - the current record pointer of any other applet shall not be changed.
- CRRN4: If the access mode is REC_ACC_MODE_PREVIOUS:
 - the previous record relative to the current selected record will be selected and updated;

- if no current record is selected, the last record will be selected and updated;
- if a cyclic EF is updated, the oldest record will be updated independent of the current record pointer and this record becomes record number 1 and the current record;
- the current record pointer of any other applet shall not be changed in case of a linear fixed EF.

6.1.1.8.1.2 Parameter errors

- CRRP1: If the currently selected EF is linear fixed and the access mode is REC_ACC_MODE_ABSOLUTE_CURRENT and recNumber is less than 0 or greater than records available, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.RECORD_NUMBER_NOT_AVAILABLE.
- CRRP2: If the currently selected EF is linear fixed and the access mode is REC_ACC_MODE_ABSOLUTE_CURRENT, recNumber is 0 and there is no current record selected, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.RECORD_NUMBER_NOT_AVAILABLE.
- CRRP3: If the currently selected EF is linear fixed and the access mode is REC_ACC_MODE_NEXT and the
 current record pointer is set to the last record, an instance of SIMViewException shall be thrown. The reason
 code shall be SIMViewException.RECORD_NUMBER_NOT_AVAILABLE.
- CRRP4: If the currently selected EF is linear fixed and the access mode is REC_ACC_MODE_PREVIOUS and the current record pointer is set to the first record; an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.RECORD_NUMBER_NOT_AVAILABLE.
- CRRP5: If the specified offset into the selected record recOffset is less than 0, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.OUT_OF_RECORD_BOUNDARIES.
- CRRP6: If recOffset plus dataLength is greater than the record length, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.OUT_OF_RECORD_BOUNDARIES.
- CRRP7: If the access mode is not between 2 and 4 inclusive (2 = REC_ACC_MODE_NEXT, etc.), an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.INVALID_MODE.
- CRRP8: If the currently selected EF is cyclic and the mode of record access mode is not REC_ACC_MODE_PREVIOUS, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.INVALID_MODE.
- CRRP9: If the array data is null, an instance of NullPointerException shall be thrown.
- CRRP10: If dataOffset is less than 0, an instance of ArrayIndexOutOfBoundsException shall be thrown.
- CRRP11: If dataLength is less than 0, an instance of ArrayIndexOutOfBoundsException shall be thrown.
- CRRP12: If dataOffset plus dataLength, is greater than the length of the array data.length, or dataOffset equals data.length, an instance of ArrayIndexOutOfBoundsException shall be thrown.

6.1.1.8.1.3 Context errors

- CRRC1: If the calling applet has currently no EF selected, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.NO_EF_SELECTED.
- CRRC2: If the currently selected EF is neither linear fixed nor cyclic, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.FILE_INCONSISTENT.
- CRRC3: If the calling applet does not fulfil the access condition, UPDATE, to perform this function, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.AC_NOT_FULFILLED.
- CRRC4: If the currently selected EF is invalidated and the file status of the EF does not allow for updating an invalidated file, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.INVALIDATION_STATUS_CONTRADICTION.

- CRRC5: If the method call causes a memory problem (e.g. memory access error), an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.MEMORY_PROBLEM.
- CRRC6: If the method call causes an error to occur that is not expected and thus not handled, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.INTERNAL_ERROR.

6.1.1.8.2 Test Suite Files

Additional requirements for the GSM personalization: This test is based on the assumption that the contents of the EFs in $DF_{SIMTEST}$ are identical to those defined in the default pre-personalization and the current record pointers have not been altered.

Test Script: API_1_SVW_UPDRSBS_BSS_1.scr

Test Applet: API_1_SVW_UPDRSBS_BSS_1.java

Load Script: API_1_SVW_UPDRSBS_BSS_1.ldr

Cleanup Script: API_1_SVW_UPDRSBS_BSS_1.clr

Parameter File: API_1_SVW_UPDRSBS_BSS_1.par

6.1.1.8.3 Test Procedure

ld	Description	API Expectation	APDU Expectation
0	SIM Initialization	Responses ignored.	
1	No EF selected recNumber = 1 mode = REC_ACC_MODE_ABSOLUTE_CURRENT recOffset = 0 byte[] data = new byte[20] dataOffset = 0 dataLength = 10 updateRecord()	Shall throw sim.access.SIMViewException with reason code NO_EF_SELECTED.	
2	<pre>Update Absolute and Current from Linear</pre>	1 - No exception shall be thrown. 2 - No exception shall be thrown. 3 - No exception shall be thrown. Resp shall be: Resp[0] = '11' Resp[1] = '11' Resp[2] = '11' Resp[3] = '11'	= 4

ld	Description	API Expectation	APDU Expectation
3	Update Current from Linear Fixed EF	1 - No exception shall be thrown.	
	1 - fid = DFSIMTEST	2 - No exception shall be thrown.	
	select()	3 - No exception shall be thrown.	
	2 - fid = EFLARU		
	select()	4 - No exception shall be thrown.	
	// Set record pointer with mode "next".	resp shall be:	
	3 - recNumber = 0	resp[0] = '22'	
	mode = REC_ACC_MODE_NEXT	resp[1] = '22'	
	recOffset = 0	resp[2] = '22'	
	data[0:3] = '00'	resp[3] = '22'	
	dataOffset = 0		
	<pre>dataLength = 4 updateRecord()</pre>		
	// write data with mode "current"		
	4 - recNumber = 0		
	data[0:3] = '22'		
	mode = REC_ACC_MODE_ABSOLUTE_CURRENT		
	updateRecord()		
	// read result with mode "absolute"		
	respOffset = 0		
	respLength = 4		
	recNumber = 1		
	mode = REC_ACC_MODE_ABSOLUTE_CURRENT		
<u> </u>	readRecord()		
4	Update Next from Linear Fixed EF, no record	1 - No exception shall be thrown.	
	pointer set	2- No exception shall be thrown.	
	1 - fid = FID_DF_SIMTEST	3 - No exception shall be thrown.	
	select()	Resp shall be:	
	2 - fid = FID_EF_LARU	Resp[0] = '33'	
	select 3 - recNumber = 0	Resp[1] = '33'	
	mode = REC_ACC_MODE_NEXT	Resp[2] = '33'	
	recOffset = 0	Resp[3] = '33'	
	data[0:3] = '33'		
	dataOffset = respOffset = 0		
	dataLength = respLength = 4		
	updateRecord()		
	mode = REC_ACC_MODE_ABSOLUTE_CURRENT		
	readRecord()		
5	Update Next from Linear Fixed EF, record	1 - No exception shall be thrown.	
	pointer set	2 - No exception shall be thrown.	
	1 - recNumber = 0	resp shall be:	
	mode = REC_ACC_MODE_NEXT	resp[0] = '44'	
	recOffset = 0 data[0:3] = '44'	resp[1] = '44'	
	dataOffset = 0	resp[2] = '44'	
	dataLength = 4	resp[3] = '44'	
	updateRecord()		
	2 - mode = REC_ACC_MODE_ABSOLUTE_CURRENT		
	readRecord()		
6	Update Next from Linear Fixed EF, no more	Shall throw	
	records	sim.access.SIMViewException with	
	recNumber = 0	reason code	
	mode = REC_ACC_MODE_NEXT	RECORD_NUMBER_NOT_AVAIL	
	recOffset = 0	ABLE.	
	data[0:3] = '55'		
	dataOffset = 0		
	<pre>dataLength = 4 updateRecord()</pre>		
7	Update Previous from Linear Fixed EF, no	1 - No exception shall be thrown.	
'	record pointer set		
	1 - fid = DFSIMTEST	2 - No exception shall be thrown.	
	-	3 - No exception shall be thrown.	
	2 - fid = EFLARU	4 - No exception shall be thrown.	
	select()	resp shall be:	
	3 - recNumber = 0	resp[0] = '66'	
	mode = REC_ACC_MODE_PREVIOUS	resp[1] = '66'	
	recOffset = 0	resp[2] = '66'	
	data[0:3] = '66'	resp[3] = '66'	
	dataOffset = respOffset = 0		
	dataLength = respLength = 4		
	updateRecord()		
	4 - mode = REC_ACC_MODE_ABSOLUTE_CURRENT		
	readRecord()	1	

ld	Description	API Expectation	APDU Expectation
8		1 - No exception shall be thrown	
	pointer set	2 - No exception shall be thrown.	
	1 - recNumber = 0	Resp shall be:	
	mode = REC_ACC_MODE_PREVIOUS	Resp[0] = '7744'	
	recOffset = 0	Resp[1] = '7744'	
	data[0:3] = '77'	Resp[2] = '7744'	
	<pre>dataOffset = respOffset = 0 dataLength = respLength = 4</pre>	Resp[3] = '7744'	
	updateRecord()		
	readRecord()		
	2 - mode = REC_ACC_MODE_ABSOLUTE_CURRENT		
9	Update Previous from Linear Fixed EF, no	Shall throw	
	more records	sim.access.SIMViewException with	
	recNumber = 0	reason code	
	mode = REC_ACC_MODE_PREVIOUS	RECORD_NUMBER_NOT_AVAIL	
	recOffset = 0	ABLE.	
	<pre>data[0:3] = '88' dataOffset = respOffset = 0</pre>		
	dataLength = respLength = 4		
	updateRecord()		
10	Update Previous from Cyclic EF	1 - No exception shall be thrown.	
	1 - fid = FID_DF_SIMTEST	2 - No exception shall be thrown.	
	select()	3 - No exception shall be thrown.	
	2 - fid = FID_EF_CARU	4 - No exception shall be thrown.	
	select()	5 - No exception shall be thrown.	
	3 - recNumber = 2 mode = REC_ACC_MODE_ABSOLUTE_CURRENT	resp shall be:	
	recOffset = 0	resp[0] = data[0]	
	respOffset = 0	resp[1] = data[1]	
	respLength = 3	resp[2] = data[2]	
	readRecord()		
	4 - recNumber = 2		
	mode = REC_ACC_MODE_PREVIOUS		
	$data[0:2] = resp[0:2] ^ 'FF'$ dataOffset = 0		
	dataLength = 3		
	updateRecord()		
	5 - recNumber = 0		
	mode = REC_ACC_MODE_ABSOLUTE_CURRENT		
	respOffset = 0		
	<pre>respLength = 3 readRecord()</pre>		
11	Update Absolute from Linear Fixed EF beyond	1 - No exception shall be thrown	
1 ''	Records	2 - Shall throw	
	1 - fid = EFLARU	sim.access.SIMViewException with	
	select()	reason code	
	2 -recNumber = -1		
	mode = REC_ACC_MODE_ABSOLUTE_CURRENT	RECORD_NUMBER_NOT_AVAIL ABLE.	
	recOffset = 0	3 - Shall throw	
	dataOffset = 0	sim.access.SIMViewException with	
	<pre>dataLength = 4 updateRecord()</pre>	reason code	
	2 - recNumber = 3	RECORD_NUMBER_NOT_AVAIL	
	updateRecord()	ABLE.	
12	No current record in linear fixed EF, update	1 - No exception shall be thrown.	
'-	current	2 - Shall throw	
	1 - fid = EFLARU	sim.access.SIMViewException with	
	select() // No curr rec	reason code	
	2 - recNumber = 0 // curr rec	RECORD_NUMBER_NOT_AVAIL	
	mode = REC_ACC_MODE_ABSOLUTE_CURRENT	ABLE.	
	recOffset = 0	, .5	
	dataOffset = 0		
	<pre>dataLength = 4 updateRecord()</pre>		
13	recOffset < 0	1 - No exception shall be thrown.	
'	1 - fid = EFLARU	2 - Shall throw	
	select()	sim.access.SIMViewException with	
	2 - recNumber = 1 // rec 1	reason code	
	mode = REC_ACC_MODE_ABSOLUTE_CURRENT	OUT_OF_RECORD_BOUNDARIE	
	recOffset = -1	S.	
	dataOffset = 0	J ^o .	
	<pre>dataLength = 4 updateRecord()</pre>		
	apaacenecora(/		

ld	Description	API Expectation	APDU Expectation
14	recOffset + dataLength > Record Length	1 - No exception shall be thrown.	•
	1 - fid = EFLARU	2 - Shall throw	
	select()	sim.access.SIMViewException with	
	2 - recNumber = 1 mode = REC_ACC_MODE_ABSOLUTE_CURRENT	reason code	
	recOffset = 2	OUT_OF_RECORD_BOUNDARIE	
	dataOffset = 0	S.	
	dataLength = 4		
	updateRecord()		
15	Updating with invalid mode	1 - No exception shall be thrown.	
	1 - fid = EFLARU select()	2 - Shall throw	
	2 - recNumber = 0	sim.access.SIMViewException with	
	mode = 1	reason code INVALID_MODE. 3 - Shall throw	
	recOffset = 0		
	dataOffset = 0	sim.access.SIMViewException with reason code INVALID_MODE.	
	<pre>dataLength = 4 updateRecord()</pre>	reason code invalib_wobe.	
	3 - mode = 5		
	updateRecord()		
16	Updating Cyclic EF with invalid mode	1 - No exception shall be thrown.	
	1 - fid = DFSIMTEST	2 - No exception shall be thrown.	
	select()	3 - Shall throw	
	2 - fid = EFCARU select()	sim.access.SIMViewException with	
	3 - recNumber = 0	reason code INVALID_MODE.	
	mode = REC_ACC_MODE_NEXT	4 - Shall throw	
	recOffset = 0	sim.access.SIMViewException with	
	data[0:2] = '00'	reason code INVALID_MODE.	
	<pre>dataOffset = 0 dataLength = 3</pre>	5 - Shall throw	
	updateRecord()	sim.access.SIMViewException with	
	4 - recNumber = 0	reason code INVALID_MODE.	
	mode = REC_ACC_MODE_ABSOLUTE_CURRENT		
	updateRecord()		
	5 - recNumber = 2 mode = REC_ACC_MODE_ABSOLUTE_CURRENT		
	updateRecord()		
17	data is null	Shall throw	
	<pre>byte[] nullBuffer = null</pre>	java.lang.NullPointerException.	
	dataOffset = 0		
	<pre>dataLength = 10 updateRecord()</pre>		
18	dataOffset < 0	Shall throw	
'0	dataOffset = -1	java.lang.	
	dataLength = 10	ArrayIndexOutOfBoundsException.	
	updateRecord()		
19	dataLength < 0	Shall throw	
	dataOffset = 0	java.lang.	
	<pre>dataLength = -1 updateRecord()</pre>	ArrayIndexOutOfBoundsException.	
20	dataOffset + dataLength > data.length	Shall throw	
-0	dataOffset = 10	java.lang.	
	dataLength = 11	ArrayIndexOutOfBoundsException.	
	updateRecord()		
21	EF is neither Cyclic nor Linear Fixed	1 - No exception shall be thrown.	
	1 - fid = DFSIMTEST	2 - No exception shall be thrown.	
	select() 2 - fid = EFTNR	3 - Shall throw	
	select()	sim.access.SIMViewException with	
	3 - dataOffset = 0	reason code	
	dataLength = 4	FILE_INCONSISTENT.	
	updateRecord()		

45

ld	Description	API Expectation	APDU Expectation
	Access condition not fulfilled 1 - fid = EFCNU select() 2 - recOffset = 0 dataOffset = 0 dataLength = 1 mode = REC_ACC_MODE_PREVIOUS updateRecord() 3 - fid = EFLNU select() 4 - recNumber = 1 mode = REC_ACC_MODE_ABSOLUTE_CURRENT recOffset = 0 dataOffset = 0 dataLength = 1 updateRecord()	1 - No exception shall be thrown. 2 - Shall throw sim.access.SIMViewException with reason code AC_NOT_FULFILLED. 3 - No exception shall be thrown. 4 - Shall throw sim.access.SIMViewException with reason code AC_NOT_FULFILLED.	
23	EF is invalidated 1 - fid = EFCNR invalidate() 2 - updateRecord() 3 - rehabilitate()	 1 - No exception shall be thrown. 2 - Shall throw sim.access.SIMViewException with reason code INVALIDATION_STATUS_CONTR ADICTION. 3 - No exception shall be thrown. 	

6.1.1.8.4 Test Coverage

CRR Number	Test Case Number	
N1	2, 3,4, 5, 7, 8, 10	
N2	2, 3	
N3	5, 6	
N4	7, 8, 9, 10	
P1	11	
P2	12	
P3	6	
P4	9	
P5	13	
P6	14	
P7	15	
P8	16	
P9	17	
P10	18	
P11	19	
P12	20	
C1	1	
C2	21	
C3	22	
C4	23	
C5, C6	Not Tested	

6.1.1.9 Method seek

Test Area Reference: API_1_SVW_SEEKB_BSS

6.1.1.9.1 Conformance Requirements

The method with the following header shall be compliant to its definition in the API.

6.1.1.9.1.1 Normal execution

- CRRN1: If the pattern in patt with the length pattLength at offset pattOffset is found in the record being specified by mode, the current record pointer is set to that record and the record number is returned. The record pointer of any other applet is not changed. This will be tested during the testing of the framework.
- CRRN2: If mode is SEEK_FROM_BEGINNING_FORWARD, the search starts with the first record forward towards the end of the file.
- CRRN3: If mode is SEEK_FROM_END_BACKWARD, the search starts with the last record backward towards the beginning of the file.
- CRRN4: If mode is SEEK_FROM_NEXT_FORWARD, the search starts from the next record after the current record pointer forward towards the end of file. If no current record pointer is selected, the search starts with the first record.
- CRRN5: If mode is SEEK_FROM_PREVIOUS_BACKWARD, the search starts from the previous record
 before the current record pointer backward towards the beginning of the file. If no current record pointer is
 selected the search starts with the last record.
- CRRN6: If pattern in patt is not found, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.PATTERN_NOT_FOUND.
- CRRN7: If mode is SEEK_FROM_NEXT_FORWARD and the record pointer is at the last record, an instance
 of SIMViewException shall be thrown. The reason code shall be
 SIMViewException.PATTERN_NOT_FOUND.
- CRRN8: If mode is SEEK_FROM_PREVIOUS_BACKWARD and the record pointer is at the first record, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.PATTERN_NOT_FOUND.

6.1.1.9.1.2 Parameter errors

- CRRP1: If mode is not between 0 and 3 inclusive (0 = SEEK_FROM_BEGINNING_FORWARD, etc.), an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.INVALID_MODE.
- CRRP2: If the pattern array patt is null, an instance of NullPointerException shall be thrown.
- CRRP3: If pattOffset is less than 0, an instance of ArrayIndexOutOfBoundsException shall be thrown.
- CRRP4: If pattLength is less than 0, an instance of ArrayIndexOutOfBoundsException shall be thrown.
- CRRP5: If pattLength is greater than the size of the record of the currently selected EF, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.OUT_OF_RECORD_BOUNDARIES.
- CRRP6: If pattOffset plus pattLength is greater than the length of the pattern array patt.length, an instance of ArrayIndexOutOfBoundsException shall be thrown.

6.1.1.9.1.3 Context errors

- CRRC1: If the calling applet has currently no EF selected, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.NO_EF_SELECTED.
- CRRC2: If the currently selected EF is not linear fixed, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.FILE_INCONSISTENT.
- CRRC3: If the calling applet does not fulfil the access condition, READ, to perform this function, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.AC_NOT_FULFILLED.
- CRRC4: If the currently selected EF is invalidated and the file status of the EF does not allow for reading an invalidated file, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.INVALIDATION_STATUS_CONTRADICTION.

- CRRC5: If the method call causes a memory problem (e.g. memory access error), an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.MEMORY_PROBLEM.
- CRRC6: If the method call causes an error to occur that is not expected and thus not handled, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.INTERNAL_ERROR.

6.1.1.9.2 Test Suite Files

Additional requirements for the GSM personalization: None

Test Script: API_1_SVW_SEEKB_BSS_1.scr

Test Applet: API_1_SVW_SEEKB_BSS_1.java

Load Script: API_1_SVW_SEEKB_BSS_1.ldr

Cleanup Script: API_1_SVW_SEEKB_BSS_1.ldr

Parameter File: API_1_SVW_SEEKB_BSS_1.par

6.1.1.9.3 Test Procedure

ld	Description	API Expectation	APDU Expectation
0	SIM Initialization	Responses ignored.	-
1	No EF selected Byte[] patt = new byte[20] pattOffset = 0	Shall throw sim.access.SIMViewException with reason code NO_EF_SELECTED.	
	<pre>pattLength = 10 mode = SEEK_FROM_BEGINNING_FORWARD seek()</pre>		
2	Pattern not Found 1 - fid = DFSIMTEST select() 2 - fid = EFLARU select() 3 - patt[0] = 'DA' pattOffset = 0 pattLength = 1 mode = SEEK_FROM_BEGINNING_FORWARD seek()	No exception shall be thrown. No exception shall be thrown. Shall throw sim.access.SIMViewException with reason code PATTERN_NOT_FOUND.	
3	Seek from Beginning Forward patt[0:2] = '55' pattOffset = 0 pattLength = 3 mode = SEEK_FROM_BEGINNING_FORWARD seek()	No exception shall be thrown. Shall return 1	
4	Seek from End Backward patt[0:2] = '55' pattOffset = 0 pattLength = 3 mode = SEEK_FROM_END_BACKWARD seek()	No exception shall be thrown. Shall return 1	
5	Seek from Next Forward patt[0:2] = 'AA' pattOffset = 0 pattLength = 3 mode = SEEK_FROM_NEXT_FORWARD seek()	No exception shall be thrown. Shall return 2	
6	Last Record, Seek from Next Forward mode = SEEK_FROM_NEXT_FORWARD seek()	Shall throw sim.access.SIMViewException with reason code PATTERN_NOT_FOUND.	
7	Seek from Previous Backward patt[0:2] = '55' pattOffset = 0 pattLength = 3 mode = SEEK_FROM_PREVIOUS_BACKWARD seek()	No exception shall be thrown. Shall return 1	

ld	Description	API Expectation	APDU Expectation
8	First Record, Seek from Previous Backward	Shall throw	P
	SEEK_FROM_PREVIOUS_BACKWARD	sim.access.SIMViewException with	
	seek()	reason code	
		PATTERN_NOT_FOUND.	
9	Pattern not Found (out of reach)	Shall throw	
	patt[0:2] = '55'	sim.access.SIMViewException with	
	<pre>pattOffset = 0 pattLength = 3</pre>	reason code	
	mode = SEEK_FROM_NEXT_FORWARD	PATTERN_NOT_FOUND.	
	seek()		
10	Invalid mode	1 - Shall throw	
	1 - mode = 4 seek()	sim.access.SIMViewException with	
	2 - mode = -1	reason code INVALID_MODE	
	seek()	2 - Shall throw sim.access.SIMViewException with	
		reason code INVALID_MODE	
11	patt is null	Shall throw	
' '	byte[] nullBuffer = null	java.lang.NullPointerException.	
	mode = SEEK_FROM_BEGINNING_FORWARD	Jaranangn tam emier = 2000 piloni	
	seek ()		
12	<pre>pattOffset < 0 patt[0:2] = '55'</pre>	Shall throw	
	pattOffset = -1	java.lang. ArrayIndexOutOfBoundsException	
	pattLength = 3	ArrayindexOdiOibodildsException	
	mode = SEEK_FROM_BEGINNING_FORWARD		
13	seek() pattLength < 0	Shall throw	
13	patt[0:2] = '55'	java.lang.	
	pattOffset = 0	ArrayIndexOutOfBoundsException	
	pattLength = -1	, aray maexe are in boarded in the	
	<pre>mode = SEEK_FROM_BEGINNING_FORWARD seek()</pre>		
14	pattLength > size of record	Shall throw	
' '	patt[0:4] = '55'	sim.access.SIMViewException with	
	pattOffset = 0	reason code	
	<pre>pattLength = 4 mode = SEEK_FROM_BEGINNING_FORWARD</pre>	OUT_OF_RECORD_BOUNDARIE	
	seek()	S	
15	pattOffset + pattLength > patt.length	Shall throw	
	patt[0:2] = '55'	java.lang.	
	<pre>pattOffset = 1 pattLength = 3</pre>	ArrayIndexOutOfBoundsException	
	mode = SEEK_FROM_BEGINNING_FORWARD		
	seek()		
16	EF is not Linear Fixed	1 - No exception shall be thrown.	
	<pre>1 - fid = EFTNU select()</pre>	2 - Shall throw	
	2 - pattOffset = 0	sim.access.SIMViewException with	
	pattLength = 3	reason code FILE_INCONSISTENT	
	mode = SEEK_FROM_BEGINNING_FORWARD	3 - Shall throw	
	seek() 3 - fid = EFCNU	sim.access.SIMViewException with	
	select()	reason code	
	seek()	FILE_INCONSISTENT	
17	Access condition not fulfilled	1 - No exception shall be thrown.	
	1 - fid = EFLNR select()	2 - Shall throw	
	2 - patt[0] = '55'	sim.access.SIMViewException with	
	pattOffset = 0	reason code AC_NOT_FULFILLED.	
	pattLength = 1	NO_NOT_I OLITELLE.	
	<pre>mode = SEEK_FROM_BEGINNING_FORWARD seek()</pre>		
18	EF is invalidated	1 - No exception shall be thrown.	
-	1 - fid = EFLARU	2 - No exception shall be thrown.	
	select()	3 - Shall throw	
	2 - invalidate() 3 - patt[0] = '55	sim.access.SIMViewException with	
	pattOffset = 0	reason code	
	pattLength = 1	INVALIDATION_STATUS_CONTR	
	mode = SEEK_FROM_BEGINNING_FORWARD	ADICTION.	
	<pre>seek() 4 - rehabilitate()</pre>	4 - No exception shall be thrown.	
	TEMANTITUACE()		

6.1.1.9.4 Test Coverage

CRR Number	Test Case Number
N1	2, 3 - 6, 7
N2	3
N3	4
N4	5
N5	7
N6	2, 6, 8, 9
N7	6
N8	8
P1	10
P2	11
P3	12
P4	13
P5	14
P6	15
C1	1
C2	16
C3	17
C4	18
C5, C6	Not Tested

6.1.1.10 Method increase

Test Area Reference: API_1_SVW_INCR_BS_BS

6.1.1.10.1 Conformance Requirements

The method with the following header shall be compliant to its definition in the API.

6.1.1.10.1.1 Normal execution

• CRRN1: The value in the array incr is added to the value of the last increased / updated record in the currently selected cyclic EF. The result is stored in the oldest record and returned in the array resp. The updated record becomes record number 1 and is selected as current record. The number of bytes of valid data in resp is returned.

6.1.1.10.1.2 Parameter errors

- CRRP1: If the array incr is null, an instance of NullPointerException shall be thrown.
- CRRP2: If incrOffset is less than 0, an instance of ArrayIndexOutOfBoundsException shall be thrown.
- CRRP3: If incrOffset plus the value 3, is greater than the length of the array incr.length, an instance of ArrayIndexOutOfBoundsException shall be thrown.
- CRRP4: If the result of the addition is greater than the maximum value of the record (represented by all bytes set to 'FF'), an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.MAX_VALUE_REACHED.
- CRRP5: If the array resp is null, an instance of NullPointerException shall be thrown.
- CRRP6: If respOffset is less than 0, an instance of ArrayIndexOutOfBoundsException shall be thrown.
- CRRP7: If the remaining length of the array resp at the offset respOffset is less than the length of the record, an instance of ArrayIndexOutOfBoundsException shall be thrown.

6.1.1.10.1.3 Context errors

- CRRC1: If the calling applet has currently no EF selected, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.NO EF SELECTED.
- CRRC2: If the currently selected EF is not cyclic, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.FILE INCONSISTENT.
- CRRC3: If increase is not allowed as indicated by the FCI byte 8 (TS 51.011: FCI structure of an EF returned by the SELECT command), an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.FILE_INCONSISTENT.
- CRRC4: If the calling applet does not fulfil the access condition, INCREASE, to perform this function, an
 instance of SIMViewException shall be thrown. The reason code shall be
 SIMViewException.AC_NOT_FULFILLED.
- CRRC5: If the currently selected EF is invalidated, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.INVALIDATION_STATUS_CONTRADICTION.
- CRRC6: If the method call causes a memory problem (e.g. memory access error), an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.MEMORY_PROBLEM.
- CRRC7: If the method call causes an error to occur that is not expected and thus not handled, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.INTERNAL_ERROR.

6.1.1.10.2 Test Suite Files

Additional requirements for the GSM personalization: None

Test Script: API_1_SVW_INCR_BS_BS_1.scr

Test Applet: API_1_SVW_INCR_BS_BS_1.java

Load Script: API_1_SVW_INCR_BS_BS_1.ldr

Cleanup Script: API_1_SVW_INCR_BS_BS_1.clr

Parameter File: API_1_SVW_INCR_BS_BS_1.par

6.1.1.10.3 Test Procedure

ld	Description	API Expectation	APDU Expectation
0	SIM Initialization	Responses ignored.	
1	<pre>No EF selected byte[] incr = new byte[4] byte[] resp = new byte[4] incrOffset = 0 respOffset = 0 increase()</pre>	Shall throw sim.access.SIMViewException with reason code NO_EF_SELECTED.	
2	<pre>Increase , verify response 1 - fid = DFSIMTEST select() 2 - fid = EFCARU select() 3 - //Set both records to 00 00 00 mode = REC_ACC_MODE_PREVIOUS data[0:3] = 0 dataOffset = 0 dataLength = 3 updateRecord() updateRecord() updateRecord() 4 - incrOffset = 0 incr[2] = 1 respOffset = 0 increase()</pre>	 1 - No exception shall be thrown. 2 - No exception shall be thrown. 3 - No exception shall be thrown. 4 - No exception shall be thrown. resp[] shall contain {0,0,1,0}. 	

ld	Description	API Expectation	APDU Expectation
3	Increase, verify file	1 - No exception shall be thrown.	-
	1 - incrOffset = 1	resp[] shall contain {0,0,0,3}.	
	<pre>incr[2] = 0, incr[3] = 2 respOffset = 1</pre>	2 - No exception shall be thrown.	
	increase()	resp[] shall contain {0,0,3,0}.	
	2 - resp[3] = 0		
	recNumber = 0		
	<pre>mode = REC_ACC_MODE_ABSOLUTE_CURRENT recOffset = 0</pre>		
	respOffset = 0		
	respLength = 0		
	readRecord()		
4	incr is null	Shall throw	
	<pre>byte[] nullBuffer = null incrOffset = 0</pre>	java.lang.NullPointerException.	
	respOffset = 0		
	increase()		
5	incrOffset < 0	Shall throw	
	<pre>incrOffset = -1 respOffset = 0</pre>	java.lang.	
	increase()	ArrayIndexOutOfBoundsException.	
6	incrOffset + 3 > incr.length	Shall throw	
	incrOffset = 2	java.lang.	
	respOffset = 0 increase()	ArrayIndexOutOfBoundsException.	
7	Reach Maximum Value	Shall throw	
'	incr[0] = incr[1] = incr[2] = 'FF'	sim.access.SIMViewException with	
	incrOffset = 0	reason code	
	respOffset = 0	MAX_VALUE_REACHED.	
8	increase() resp is null	Shall throw	
0	incr[0] = incr[1] = 0x00'	java.lang.NullPointerException.	
	incr[2] = '02'	javanang.rvam emterzxoeption.	
	incrOffset = 0		
	<pre>byte[] respNull = null respOffset = 0</pre>		
	increase()		
9	respOffset < 0	Shall throw	
	incrOffset = 0	java.lang.	
	respOffset = -1 increase()	ArrayIndexOutOfBoundsException.	
10	respOffset + recordLength > resp.length	Shall throw	
. •	incrOffset = 0	java.lang.	
	respOffset = 2	ArrayIndexOutOfBoundsException.	
44	increase()	4. No exception shall be through	
11	EF is not Cyclic 1 - fid = EFTARU	1 - No exception shall be thrown.2 - Shall throw	
	select()	sim.access.SIMViewException with	
	2 - incrOffset = 0	reason code	
	respOffset = 0 increase()	FILE_INCONSISTENT.	
	3 - fid = EFLARU	3 - No exception shall be thrown.	
	select()	4 - Shall throw	
	4 - incrOffset = 0	sim.access.SIMViewException with	
	<pre>respOffset = 0 increase()</pre>	reason code	
12	Access condition not fulfilled	FILE_INCONSISTENT. 1 - No exception shall be thrown.	
12	1 - fid = EFCNIC	2 - Shall throw	
	select()	sim.access.SIMViewException with	
	2 - incrOffset = 0	reason code	
	<pre>respOffset = 0 increase()</pre>	AC_NOT_FULFILLED.	
13	EF is invalidated	1 - No exception shall be thrown.	
	1 - fid = EFCARU	2 - No exception shall be thrown.	
	select()	3 - Shall throw	
	<pre>2 - invalidate() 3 - incrOffset = 0</pre>	sim.access.SIMViewException with	
	respOffset = 0	reason code	
	increase()	INVALIDATION_STATUS_CONTR	
	4 - rehabilitate()	ADICTION.	
		4 - No exception shall be thrown.	

ld	Description	API Expectation	APDU Expectation
14	Check increase not allowed from FCI	1 - No exception shall be thrown.	
		Bit 7 of resp[7] shall not be set (0),	
	fciLength = 8	indicating that increase is not	
	select (FID_EF_CINA, fci)	allowed.	
	<pre>Verify FCI byte 8 (fci[7]) 2 - incrOffset = 0</pre>	2 - Shall throw	
	respOffset = 0	sim.access.SIMViewException with	
		reason code	
		FILE_INCONSISTENT	

6.1.1.10.4 Test Coverage

CRR Number	Test Case Number
N1	2, 3
P1	4
P2	5
P3	6
P4	7
P5	8
P6	9
P7	10
C1	1
C2	11
C3	14
C4	12
C5	13
C6, C7	Not Tested

6.1.1.11 Method invalidate

Test Area Reference: API_1_SVW_INVL

6.1.1.11.1 Conformance Requirements

The method with the following header shall be compliant to its definition in the API.

6.1.1.11.1.1 Normal execution

• CRRN1: The currently selected EF of the calling applet shall be invalidated, i.e. the flag in the EF file status shall be changed accordingly.

6.1.1.11.1.2 Parameter errors

No requirements.

6.1.1.11.1.3 Context errors

- CRRC1: If the calling applet has currently no EF selected, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.NO_EF_SELECTED.
- CRRC2: If the calling applet does not fulfil the access condition, INVALIDATE, to perform this function, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.AC_NOT_FULFILLED.
- CRRC3: If the currently selected EF is already invalidated, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.INVALIDATION_STATUS_CONTRADICTION.

- CRRC4: If the method call causes a memory problem (e.g. memory access error), an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.MEMORY_PROBLEM.
- CRRC5: If the method call causes an error to occur that is not expected and thus not handled, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.INTERNAL_ERROR.

6.1.1.11.2 Test Suite Files

Additional requirements for the GSM personalization: None

Test Script: API_1_SVW_INVL_1.scr

Test Applet: API_1_SVW_INVL_1.java

Load Script: API_1_SVW_INVL_1.ldr

Cleanup Script: API_1_SVW_INVL_1.clr

Parameter File: API_1_SVW_INVL_1.par

6.1.1.11.3 Test Procedure

ld	Description	API Expectation	APDU Expectation
0	SIM Initialization	Responses ignored.	
1	No EF is selected	1 - Shall throw	
	1 - invalidate()	sim.access.SIMViewException with	
		reason code NO_EF_SELECTED.	
2	Invalidate EF	1 - No exception shall be thrown.	
	1 - fid = DFSIMTEST	2 - No exception shall be thrown.	
	select()	3 - No exception shall be thrown.	
	2 - fid = EFTNR	4 - No exception shall be thrown.	
	select()	· · · · · · · · · · · · · · · · · · ·	
	3 - invalidate()		
	4 - rehabilitate()	4 1 1 11 1	
3	Access condition not fulfilled	1 - No exception shall be thrown.	
	1 - fid = EFCNIV	2 - Shall throw	
	<pre>select() 2 - invalidate()</pre>	sim.access.SIMViewException with	
	2 - Invalidate()	reason code	
		AC_NOT_FULFILLED.	
4	EF is already invalidated	 No exception shall be thrown. 	
	1 - fid = EFTNR	2 - No exception shall be thrown.	
	select()	3 - Shall throw	
	2 - invalidate()	sim.access.SIMViewException with	
	3 - invalidate()	reason code	
		INVALIDATION STATUS CONTR	
		ADICTION.	

6.1.1.11.4 Test Coverage

CRR number	Test Case Number
N1	2
C1	1
C2	3
C3	4
C4, C5	Not Tested

6.1.1.12 Method rehabilitate

Test Area Reference: API_1_SVW_REHA

6.1.1.12.1 Conformance Requirements

The method with the following header shall be compliant to its definition in the API.

6.1.1.12.1.1 Normal execution

• CRRN1: The currently selected EF of the calling applet shall be rehabilitated, i.e. the flag in the EF file status shall be changed accordingly.

6.1.1.12.1.2 Parameter errors

No requirements.

6.1.1.12.1.3 Context errors

- CRRC1: If the calling applet has currently no EF selected, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.NO_EF_SELECTED.
- CRRC2: If the calling applet does not fulfil the access condition, REHABILITATE, to perform this function, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.AC_NOT_FULFILLED.
- CRRC3: If the currently selected EF is not invalidated, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.INVALIDATION_STATUS_CONTRADICTION.
- CRRC4: If the method call causes a memory problem (e.g. memory access error), an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.MEMORY_PROBLEM.
- CRRC5: If the method call causes an error to occur that is not expected and thus not handled, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.INTERNAL_ERROR.

6.1.1.12.2 Test Suite Files

Additional requirements for the GSM personalization: None

Test Script: API_1_SVW_REHA_1.scr

Test Applet: API_1_SVW_REHA_1.java

Load Script: API_1_SVW_REHA_1.ldr

Cleanup Script: API_1_SVW_REHA_1.clr

Parameter File: API_1_SVW_REHA_1.par

6.1.1.12.3 Test Procedure

ld	Description	API Expectation	APDU Expectation
0	SIM Initialization	Responses ignored.	
1	No EF is selected	1 - Shall throw	
	1 - rehabilitate()	sim.access.SIMViewException with reason code NO EF SELECTED.	

ld	Description	API Expectation	APDU Expectation
2	Rehabilitate invalidated File 1 - fid = DFSIMTEST select() 2 - fid = EFCNR select() 3 - invalidate() 4 - rehabilitate() 5 - byte[] incr = new byte[3] = {0,0,1} incrOffset = 0 byte[] resp = new byte[1] = 1 respOffset = 0 increase()	1 - No exception shall be thrown. 2 - No exception shall be thrown. 3 - No exception shall be thrown. 4 - No exception shall be thrown. 5 - No exception shall be thrown. resp[] shall contain {0,0,1}.	·
3	Access condition not fulfilled 1 - fid = EFCNRH select() 2 - rehabilitate()	1 - No exception shall be thrown. 2 - Shall throw sim.access.SIMViewException with reason code AC_NOT_FULFILLED	
4	Rehabilitate validated File 1 - fid = EFCNR select() 2 - rehabilitate()	1 - No exception shall be thrown. 2 - Shall throw sim.access.SIMViewException with reason code INVALIDATION_STATUS_CONTR ADICTION.	

6.1.1.12.4 Test Coverage

CRR number	Test Case Number
N1	2
C1	1
C2	3
C3	4
C4, C5	Not Tested

6.1.2 Class SIMSystem

6.1.2.1 Method getTheSIMView

Test Area Reference: API_1_SSY_GETS

6.1.2.1.1 Conformance Requirement:

The method with following header shall compliant to its definition in the API.

public static SIMView getTheSIMView()

6.1.2.1.1.1 Normal execution

• CRRN1: returns a reference to class which implements the SIMView interface.

6.1.2.1.1.2 Parameters error

No requirements.

6.1.2.1.1.3 Context errors

No requirements.

6.1.2.1.2 Test suite files

No additional requirements for the GSM personalization:

Test Script: API_1_SSY_GETS_1.scr

Test Applet: API_1_SSY_GETS_1.java

Load Script: API_1_SSY_GETS_1.ldr

Cleanup Script: API_1_SSY_GETS_1.clr

Parameter File: API_1_SSY_GETS_1.par

6.1.2.1.3 Test Procedure

ld	Description	API Expectation	APDU Expectation
1	reference not equal null after execute	The returned reference shall be not	
		null after execute	
2	reference to the GSM interface	Returned a reference to the GSM	
		interface	

6.1.2.1.4 Test Coverage

CRR number	Test case number
N1	1,2

6.1.3 Class SIMViewException

6.1.3.1 Method throwIt

Test Area Reference: API_1_SVE_THITS

6.1.3.1.1 Conformance Requirement:

The method with following header shall be compliant to its definition in the API.

6.1.3.1.1.1 Normal execution

- CRRN1: Throws the JCRE instance of SIMViewException with the specified reason.
- CRRN2: Extends javacard.framework.CardRuntimeException.

6.1.3.1.1.2 Parameter errors

No requirements.

6.1.3.1.1.3 Context errors

No requirements.

6.1.3.1.2 Test Suite Files

No additional requirements for the GSM personalization

Test Script: API_1_SVE_THITS_1.scr

Test Applet: API_1_SVE_THITS_1.java

Load Script: API_1_SVE_THITS_1.ldr

Cleanup Script: API_1_SVE_THITS_1.clr

Parameter File: API_1_SVE_THITS_1.par

6.1.3.1.3 Test Procedure

ld	Description	API Expectation	APDU Expectation
1	Throws the JCRE instance of	Reason = 0	
	SIMViewException with the specified reason		
2	Throws the JCRE instance of	Reason = 1	
	SIMViewException with the specified reason		
3	Throws the JCRE instance of	Reason = 15	
	SIMViewException with the specified reason		
4	SIMViewException extends	Reason = 0	
	javacard.framework.CardRuntimeException		
5	SIMViewException extends	Reason = 1	
	javacard.framework.CardRuntimeException		
6	SIMViewException extends	Reason = 15	
	javacard.framework.CardRuntimeException		

6.1.3.1.4 Test Coverage

CRR number	Test case number	
N1	1,2,3	
N2	4,5,6	

6.1.3.2 Constructor

Test Area Reference: API_1_SVE_COORS

6.1.3.2.1 Conformance Requirement:

The method with following header shall compliant to its definition in the API.

6.1.3.2.1.1 Normal execution

• CRRN1: Construct a SIMViewException with the specified reason.

6.1.3.2.1.2 Parameters error

No requirements.

6.1.3.2.1.3 Context errors

No requirements.

6.1.3.2.2 Test suite files

No additional requirements for the GSM personalization

Test Script: API_1_SVE_COORS_1.scr

Test Applet: API_1_SVE_COORS_1.java

Load Script: API_1_SVE_COORS.ldr

Cleanup Script: API_1_SVE_COORS.clr

Parameter File: API_1_SVE_COORS.par

6.1.3.2.3 Test Procedure

ld	Description	API Expectation	APDU Expectation
1	SIMViewException with the specified reason	Reason (specified)	
	(The reason shall set with setReason and		
	compare the Exception with getReason)		

6.1.3.2.4 Test Coverage

CRR number	Test case number
N1	1

6.1.3.3 Reason Codes

Test Area Reference: API_1_SVE_CONS

6.1.3.3.1 Conformance Requirement:

There is no API, only constants. This constants shall compliant to its definition in the API.

6.1.3.3.1.1 Normal execution

- CRRN1: The Constants of the class SIMViewException shall all have the same name and value defined in the 3GPP TS 43.019 [7].
- CRRN2: Constructs SIMViewException a Exception with the specified reason.

6.1.3.3.1.2 Parameters error

No requirements.

6.1.3.3.1.3 Context errors

No requirements.

6.1.3.3.2 Test suite files

None.

6.1.3.3.3 Test Procedure

The constants in Java are resolved at compilation time, therefore a runtime test is not useful. No test of constants will be performed

6.2 Package sim.toolkit

6.2.1 Interface ToolkitConstants

6.2.1.1 Constants

Test Area Reference: API_2_TKC_CONS

6.2.1.1.1 Conformance Requirement

There is no API, only constants. This constants shall be compare to its definition in the API.

6.2.1.1.1.1 Normal execution

• CRRN1: The Toolkit Constants shall all have the same name and value as defined in 3GPP TS 43.019 [7].

6.2.1.1.1.2 Parameters error

No requirements.

6.2.1.1.3 Context errors

No requirements.

6.2.1.1.2 Test suite files

None.

6.2.1.1.3 Test Procedure

The constants in Java are resolved at compilation time, therefore a runtime test is not useful. No test of constants will be performed.

6.2.2 Interface ToolkitInterface

6.2.2.1 Method processToolkit

Test Area Reference: API_2_TKI_PRTKB

6.2.2.1.1 Conformance Requirement:

The method with following prototype shall be compliant to its definition in the API.

6.2.2.1.1.1 Normal execution

- CRRN1: This interface must be implemented by a Toolkit applet (which extends the javacard.framework.Applet class) so that it can be triggered by the Toolkit Handler according to the registration information.
- CRRN2: The Toolkit applet will have to implement the processToolkit shared method so that the following events can be notified:

Event	Description	
EVENT_PROFILE_DOWNLOAD	Terminal Profile command reception	
EVENT_FORMATTED_SMS_PP_ENV	Formatted envelope SMS-PP Data Download	
	reception	
EVENT_FORMATTED_SMS_PP_UPD	Formatted Update Record EF SMS	
EVENT_FORMATTED_SMS_CB	Formatted envelope Cell Broadcast Data	
	Download command reception	
EVENT_UNFORMATTED_SMS_PP_ENV	Unformatted Envelope SMS-PP Data Download	
	reception	
EVENT_UNFORMATTED_SMS_PP_UPD	Unformatted Update Record EF SMS	
EVENT_UNFORMATTED_SMS_CB	Unformatted Cell Broadcast Data Download	
	command reception	
EVENT_MENU_SELECTION	Envelope Menu Selection command reception	
EVENT_MENU_SELECTION_HELP_REQUEST	Envelope Menu Selection Help Request	
	command reception	
EVENT_CALL_CONTROL_BY_SIM	Envelope Call Control by SIM command reception	

Event	Description	
EVENT_MO_SHORT_MESSAGE_CONTROL_BY_SIM	Envelope MO Short Message Control by SIM	
	command reception	
EVENT_TIMER_EXPIRATION	Envelope Timer Expiration	
EVENT_EVENT_DOWNLOAD_MT_CALL	Envelope Event Download - MT call	
EVENT_EVENT_DOWNLOAD_CALL_CONNECTED	Envelope Event Download - Call connected	
EVENT_EVENT_DOWNLOAD_CALL_DISCONNECTED	Event Download - Call disconnected	
EVENT_EVENT_DOWNLOAD_LOCATION_STATUS	Envelope Event Download - Location status	
EVENT_EVENT_DOWNLOAD_USER_ACTIVITY	Envelope Event Download - User activity	
EVENT_EVENT_DOWNLOAD_IDLE_SCREEN_AVAILABLE	Envelope Event Download - Idle screen available	
EVENT_EVENT_DOWNLOAD_CARD_READER_STATUS	Envelope Event Download - Card Reader Status	
EVENT_EVENT_DOWNLOAD_LANGUAGE_SELECTION	Envelope Event Download - Language Selection	
EVENT_EVENT_DOWNLOAD_BROWSER_TERMINATION	Envelope Event Download - Browser Termination	
EVENT_EVENT_DOWNLOAD_DATA_AVAILABLE	Envelope Event Download - Data Available	
EVENT_EVENT_DOWNLOAD_CHANNEL_STATUS	Envelope Event Download - Channel Status	
EVENT_FIRST_COMMAND_AFTER_SELECT	First command performed after select GSM	
	application or ATR	
EVENT_STATUS_COMMAND	Status APDU command event	
EVENT_UNRECOGNIZED_ENVELOPE	Unrecognized Envelope command reception	

6.2.2.1.1.2 Parameters error

No requirements.

6.2.2.1.1.3 Context errors

No requirements.

6.2.2.1.2 Test suite files

The method is tested in the Framework.

6.2.2.1.3 Test Coverage

CRR number Test case number	
N1	Tested in Framework
N2 Tested in Frameworl	

6.2.3 Class EditHandler

It is not possible to test the methods provided by this class as it is declared 'abstract'; it will be done in the class inheriting it: EnvelopeResponseHandler, ProactiveHandler.

6.2.4 Class EnvelopeHandler

6.2.4.1 Method getEnvelopeTag

Test Area Reference: API_2_ENH_GENT

6.2.4.1.1 Conformance Requirement:

The method with following header shall be compliant to its definition in the API.

public byte getEnvelopeTag()

6.2.4.1.1.1 Normal execution

• CRRN1: The method shall return the Envelope BER-TLV tag.

• CRRN2: The Envelope BER TAG is available for all triggered toolkit applets from the invocation to the termination of their processToolkit method if the EnvelopeHandler is available.

6.2.4.1.1.2 Parameters error

No requirements.

6.2.4.1.1.3 Context errors

No requirements.

6.2.4.1.2 Test suite files

Test Script: API_2_ENH_GENT_1.scr

Test Applet: API_2_ENH_GENT_1.java

Load Script: API_2_ENH_GENT_1.ldr

Cleanup Script: API_2_ENH_GENT_1.clr

Parameter File: API_2_ENH_GENT_1.par

6.2.4.1.3 Test Procedure

ld	Description	API Expectation	APDU Expectation
	getEnvelopeTag called just after triggering of the application.	Returns 0xD1	
	getEnvelopeTag called after a proactive command.	Returns 0xD1	
	getEnvelopeTag called after a second proactive command.	Returns 0xD1	

6.2.4.1.4 Test Coverage

CRR number	Test case number
N1	1, 2, 3
N2	1, 2, 3

6.2.4.2 Method getItemIdentifier

Test Area Reference: API_2_ENH_GIID

6.2.4.2.1 Conformance Requirement:

The method with following header shall be compliant to its definition in the API.

6.2.4.2.1.1 Normal execution

- CRRN1:The method shall return the item identifier byte value.
- CRRN2:The item identifier byte value returned shall be from the first Item Identifier TLV element.
- CRRN3: If the element is available it becomes the TLV selected.
- CRRN4: The item identifier is available for all triggered toolkit applets from the invocation to the termination of their processToolkit method if the EnvelopeHandler is available.

6.2.4.2.1.2 Parameters error

No requirements.

6.2.4.2.1.3 Context errors

- CRRC1: The method shall throw ToolkitException (UNAVAILABLE_ELEMENT) if the item identifier TLV is not present.
- CRRC2: The method shall throw ToolkitException (OUT_OF_TLV_BOUNDARIES) if the item identifier byte is missing in the Item Identifier Simple TLV.

6.2.4.2.2 Test suite files

Test Script: API_2_ENH_GIID_1.scr

Test Applet: API_2_ENH_GIID_1.java

Load Script: API_2_ENH_GIID_1.ldr

Cleanup Script: API_2_ENH_GIID_1.clr

Parameter File: API_2_ENH_GIID_1.par

6.2.4.2.3 Test Procedure

ld	Description	API Expectation	APDU Expectation
1	Send envelope SMS-PP Formatted with item	Returns 03	
	identifier TLV and identifier value of 03		
2	Send envelope SMS-PP Formatted with two item	Returns FF	
	identifier TLV with first value FF and second 44		
3	Send envelope SMS-PP Formatted with two item	Returns 81	
	identifier TLV with first value 81 and second 44,	Returns 81	
	call twice the method getItemIdentifier		
4	Send envelope SMS-PP Formatted with item	getItemIdentifier=getValueByte	
	identifier TLV and value of 66. FindTLV with TAG		
	02. getItemIdentifier and then getValueByte with		
	offset 0		
5	Send envelope SMS-PP Formatted without item	ToolkitException	
	identifier TLV and getItemIdentifier	(UNAVAILABLE_ELEMENT)	
6	Send Envelope SMS-PP Formatted with item	Returns 66	
	identifier TLV (66), send proactive command. Then		
	getItemIdentifier		
7	Send Envelope SMS-PP Formatted with item	ToolkitException	
	identifier TLV but without item number	(OUT_OF_TLV_BOUNDARIES)	

6.2.4.2.4 Test Coverage

CRR number	Test case number	
N1	1, 2, 3	
N2	2, 3	
N3	4	
N4	6	
C1	5	
C2	7	

6.2.4.3 Method getSecuredDataLength

Test Area Reference: API_2_ENH_GSDL

6.2.4.3.1 Conformance Requirement:

The method with following header shall be compliant to its definition in the API.

6.2.4.3.1.1 Normal execution

- CRRN1: The method shall return the length of the Secured Data from the Command Packet in the SMS TPDU (simple or concatenated) or Cell Broadcast Page Simple TLV contained in the Envelope handler.
- CRRN2: The length is from the first SMS TPDU TLV or Cell Broadcast Page Simple TLV.
- CRRN3: The length should not include padding bytes.
- CRRN4: The method can be used if the event is EVENT_FORMATTED_SMS_PP_ENV and if the SMS TP-UD is formatted according to 3GPP TS 23.048 [8].
- CRRN5: The method can be used if the event is EVENT_FORMATTED_SMS_PP_UPD and if the SMS TP-UD is formatted according to 3GPP TS 23.048 [8].
- CRRN6: The method can be used if the event is EVENT_FORMATTED_SMS_CB and if the Cell Broadcast Page is formatted according to 3GPP TS 23.048 [8].
- CRRN7: If the method is successful and if the event is EVENT_FORMATTED_SMS_PP_ENV, the selected TLV should be the SMS TPDU TLV.
- CRRN8: If the method is successful and if the event is EVENT_FORMATTED_SMS_PP_UPD, the selected TLV should be the SMS TPDU TLV.
- CRRN9: If the method is successful and if the event is EVENT_FORMATTED_SMS_CB, the selected TLV should be the Cell Broadcast Page TLV.

6.2.4.3.1.2 Parameters error

No requirements.

6.2.4.3.1.3 Context errors

- CRRC1: The method shall thrown ToolkitException (UNAVAILABLE_ELEMENT) in case of unavailable SMS TPDU TLV element or Cell Broadcast Page Simple TLV.
- CRRC2: The method shall thrown ToolkitException (UNAVAILABLE_ELEMENT) in case of wrong data format.

6.2.4.3.2 Test suite files

Specific triggering:

- FORMATTED SMS CB.
- UNFORMATTED SMS CB.
- FORMATTED SMS PP UPD.
- UNFORMATED SMS PP ENV.
- For Formatted triggering if CC/RC/DS is used, the security parameters are the one used for downloading applications.

Test Script: API_2_ENH_GSDL_1.scr

Test Applet: API_2_ENH_GSDL_1.java

Load Script: API_2_ENH_GSDL_1.ldr

Cleanup Script: API_2_ENH_GSDL_1.clr

Parameter File: API_2_ENH_GSDL_1.par

6.2.4.3.3 Test procedure

ld	Description	API Expectation	APDU Expectation
	FORMATTED SMS PP ENV Triggering		

ld	Description	API Expectation	APDU Expectation
1	Description Test with FORMATTED_SMS_PP_ENV and TP-	Returns 0x002A	APDO Expectation
'	OA length of 2	Returns 0x002A	
2	Test with TP-OA length of 6	Returns 0x002A	
3	Test with TP-OA length of 12	Returns 0x002A	
4	Test with RC/CC/DS length of 0	Returns 0x0010	
5	Test with RC/CC/DS length of 8	Returns 0x0010	
6	Test with PCNTR = 0	Returns 0x0010	
7	Test with PCNTR = 7	Returns 0x0005	
8	Test with Secured Data Length = 00	Returns 0x0000	
9	Test with Secured Data Length = 0x33	Returns 0x0033	
10	Test with Secured Data Length = 0x6C (UDL = 0x7F)	Returns 0x006C	
11	Test with Secured Data Length = 0x6D (UDL = 0x80)	Returns 0x006D	
12	Test with Secured Data Length = maximum length for one envelope : 0x79 (UDL = 0x8C)	Returns 0x0079	
13	Verify it is the first TPDU TLV: Send a SMS PP with 2 TPDU TLV and inside two different secured data lengths: 5 and 10	Returns 0x0005	
14	Test with secured data length = 0x7F (2 concatenated envelopes are needed)	Returns 0x007F	
	Test with secured data length = 0x80 (2 concatenated envelopes are needed)	Returns 0x0080	
	Test with secured data length = maximum length for 2 concatenated envelopes : 0xFA	Returns 0x00FA	
17	Test with FORMATTED_SMS_PP_ENV Verify after call of the method the current TLV is the TPDU TLV:	getValueByte returns 0x0040	
	findTLV device identities, getSecuredDataLength and then getValueByte to verify that the current TLV is the TPDU TLV		
	FORMATTED SMS PP UPD Triggering		
18	Same test as 1 but with FORMATTED_SMS_PP_UPD	Returns 0x002A	
19	Same test as 2 but with FORMATTED_SMS_PP_UPD	Returns 0x002A	
20	Same test as 3 but with FORMATTED_SMS_PP_UPD	Returns 0x002A	
	Same test as 4 but with FORMATTED_SMS_PP_UPD	Returns 0x0010	
	Same test as 5 but with FORMATTED_SMS_PP_UPD	Returns 0x0010	
23	Same test as 6 but with FORMATTED_SMS_PP_UPD	Returns 0x0010	
	Same test as 7 but with FORMATTED_SMS_PP_UPD	Returns 0x0005	
	Same test as 8 but with FORMATTED_SMS_PP_UPD	Returns 0x0000	
	Same test as 9 but with FORMATTED_SMS_PP_UPD	Returns 0x0033	
	Same test as 10 but with FORMATTED_SMS_PP_UPD	Returns 0x006C	
	Same test as 11 but with FORMATTED_SMS_PP_UPD	Returns 0x006D	
	Same test as 12 but with FORMATTED_SMS_PP_UPD	Returns 0x0079	
	Same test as 13 but with FORMATTED_SMS_PP_UPD	Returns 0x0005	
31	Test with secured data length = 0x7F (2 concatenated envelopes are needed)	Returns 0x007F	
	Test with secured data length = 0x80 (2 concatenated envelopes are needed)	Returns 0x0080	
	Test with secured data length = maximum length for 2 concatenated envelopes : 0xFA	Returns 0x00FA	
34	Test with FORMATTED_SMS_PP_UPD Verify after call of the method the current TLV is	getValueByte returns 0x0040	

ld	Description	API Expectation	APDU Expectation
	the TPDU TLV:		
	findTLV device identities, getSecuredDataLength		
	and then getValueByte to verify that the current		
	TLV is the TPDU TLV		
	FORMATTED SMS CB Triggering		
35	Same test as 4 but with FORMATTED_SMS_CB	Returns 0x0010	
36	Same test as 5 but with FORMATTED_SMS_CB	Returns 0x0010	
37	Same test as 6 but with FORMATTED_SMS_CB	Returns 0x0010	
38	Same test as 7 but with FORMATTED_SMS_CB	Returns 0x0005	
39	Same test as 8 but with FORMATTED_SMS_CB	Returns 0x0000	
40	Same test as 9 but with FORMATTED_SMS_CB	Returns 0x0033	
41	Same test as 12 but with maximum secured data	Returns 0x0042	
	length: 0x42, and FORMATTED_SMS_CB		
42	Test with FORMATTED_SMS_CB	getValueByte returns 0x00	
	Verify after call of the method the current TLV is		
	the Cell Broadcast Page TLV:		
	findTLV device identities, getSecuredDataLength		
	and then getValueByte to verify that the current		
	TLV is the Cell Broadcast Page TLV		
	Error tests		
43	Send an envelope SMS CB,	ToolkitException	
	getSecuredDataLength	UNAVAILABLE_ELEMENT	
44	Send an envelope SMS PP unformatted	ToolkitException	
		UNAVAILABLE_ELEMENT	

6.2.4.3.4 Test Coverage

CRR number	Test case number	
N1	1 to 42	
N2	13, 30	
N3	6, 7, 23, 24, 37, 38	
N4	1 to 17	
N5	18 to 34	
N6	35 to 42	
N7	17	
N8	34	
N9	42	
C1	43	
C2	44	

6.2.4.4 Method getSecuredDataOffset

Test Area Reference: API_2_ENH_GSDO

6.2.4.4.1 Conformance Requirement:

The method with following header shall be compliant to its definition in the API.

6.2.4.4.1.1 Normal execution

- CRRN1: The method shall return the offset of the secured data first byte contained in a SMS TPDU TLV.
- CRRN2: The offset is from the first SMS TPDU TLV.

- CRRN3: The method can be used if the event is EVENT_FORMATTED_SMS_PP_ENV and if the SMS TP-UD is formatted according to 3GPP TS 23.048 [8].
- CRRN4: The method can be used if the event is EVENT_FORMATTED_SMS_PP_UPD and if the SMS TP-UD is formatted according to 3GPP TS 23.048 [8].
- CRRN5: The method can be used if the event is EVENT_FORMATTED_SMS_CB and if the Cell Broadcast Page is formatted according to 3GPP TS 23.048 [8].
- CRRN6: If the method is successful and if the event is EVENT_FORMATTED_SMS_PP_ENV, the selected TLV should be the SMS TPDU TLV.
- CRRN7: If the method is successful and if the event is EVENT_FORMATTED_SMS_PP_UPD, the selected TLV should be the SMS TPDU TLV.
- CRRN8: If the method is successful and if the event is EVENT_FORMATTED_SMS_CB, the selected TLV should be the Cell Broadcast Page TLV.
- CRNN9: If the Secured Data length is zero the value returned shall be the offset of the first byte following the 3GPP TS 23.048 [8] Command Packet structure.

6.2.4.4.1.2 Parameters error

No requirements.

6.2.4.4.1.3 Context errors

- CRRC1: The method shall thrown ToolkitException (UNAVAILABLE_ELEMENT) in case of unavailable SMS TPDU TLV element.
- CRRC2: The method shall thrown ToolkitException (UNAVAILABLE_ELEMENT) in case of wrong data format.

6.2.4.4.2 Test suite files

Specific triggering:

- FORMATTED SMS CB.
- UNFORMATTED SMS CB.
- FORMATTED SMS PP UPD.
- UNFORMATED SMS PP ENV.
- For Formatted triggering if CC/RC/DS is used, the security parameters are the one used for downloading applications.

Test Script: API_2_ENH_GSDO_1.scr

Test Applet: API_2_ENH_GSDO_1.java

Load Script: API_2_ENH_GSDO_1.ldr

Cleanup Script: API_2_ENH_GSDO_1.clr

Parameter File: API_2_ENH_GSDO_1.par

6.2.4.4.3 Test Procedure

	ld	Description	API Expectation	APDU Expectation
ĺ		FORMATTED SMS PP ENV triggering		

ld	Description	API Expectation	APDU Expectation
1	Test with TP-OA length of 2 and RC/CC/DS length	Returns 0x21	-
	is 0		
2	Test with TP-OA length of 6 and RC/CC/DS length	Returns 0x23	
-	is 0		
3	Test with TP-OA length of 12 and RC/CC/DS	Returns 0x26	
"	length is 0	TROTAINS ONEO	
4	Test with RC/CC/DS length of 0 and TP-OA length	Poturne 0v21	
4	is 2	Returns 0x2 i	
_		Datuma 0x00	
5	Test with RC/CC/DS length of 8 and TP-OA length	Returns 0x29	
	is 2	D	
6	Send a SMS PP with 2 TPDU TLV and inside two	Returns 0x24 (the first offset)	
	different secured data offsets		
	Same test as 1 but without any secured data	Returns 0x21	
8	Test with FORMATTED_SMS_PP ENV	Returns 0x40	
	Verify after call of the method the current TLV is		
	the TPDU TLV:		
	findTLV device identities, getSecuredDataOffset		
	and then getValueByte to verify that the current		
	TLV is the TPDU TLV		
9	Same test as 1, but with a concatenated SMS (2	Returns 0x21	
	Short Messages and maximum Secured Data		
	Length = $0x00FA$)		
	FORMATTED SMS PP UPR triggering		
10	Same test as 1 but with	Returns 0x21	
	FORMATTED_SMS_PP_UPD		
11	Same test as 2 but with	Returns 0x23	
' '	FORMATTED_SMS_PP_UPD		
12	Same test as 3 but with	Returns 0x26	
'-	FORMATTED_SMS_PP_UPD	TOTALING GAZO	
13	Same test as 4 but with	Returns 0x21	
'0	FORMATTED_SMS_PP_UPD	TOTALIS ONE I	
1/	Same test as 5 but with	Returns 0x29	
'-	FORMATTED_SMS_PP_UPD	INCIGITIS OXES	
15	Same test as 6 but with	Returns 0x24 (the first offset)	
15		Returns 0x24 (the first offset)	
16	FORMATTED_SMS_PP_UPD Same test as 7 but with	Returns 0x21	
10		Returns 0x2 i	
47	FORMATTED_SMS_PP_UPD	D = t 0 - 40	
17	Test with FORMATTED_SMS_PP UPD	Returns 0x40	
	Verify after call of the method the current TLV is		
	the TPDU TLV:		
	findTLV device identities, getSecuredDataOffset		
	and then getValueByte to verify that the current		
40	TLV is the TPDU TLV	D	
18	Same test as 10, but with a concatenated SMS (2	Returns 0x21	
	Short Messages and maximum Secured Data		
	Length = 0x00FA)		
	FORMATTED SMS CB triggering	D	
	Same test as 4 but with FORMATTED_SMS_CB	Returns 0x16	
	Same test as 5 but with FORMATTED_SMS_CB	Returns 0x1E	
	Same test as 7 but with FORMATTED_SMS_CB	Returns 0x16	
22	Test with FORMATTED_SMS_CB	Returns 0x00	
	Verify after call of the method the current TLV is		
	the Cell Broadcast Page TLV:		
	findTLV device identities, getSecuredDataOffset		
	and then getValueByte to verify that the current		
	TLV is the Cell Broadcast Page TLV		
	UNFORMATTED Triggering		
23	Send an UNFORMATTED SMS CB envelope,	ToolkitException	
	getSecuredDataOffset	UNAVAILABLE_ELEMENT	
24	Send an UNFORMATTED SMS PP envelope,	ToolkitException	
	getSecuredDataOffset	UNAVAILABLE_ELEMENT	
	(g - · · · · · · · · · · · · · · ·		

6.2.4.4.4 Test Coverage

CRR number	Test case number	
N1	1 to 22.	
N2	6, 15.	
N3	1 to 9.	
N4	10 to 18.	
N5	19, 20, 21, 22	
N6	8	
N7	17	
N8	22	
N9	7, 16, 21.	
C1	23	
C2	24	

6.2.4.5 Method getTheHandler

Test Area Reference: API_2_ENH_GTHD

6.2.4.5.1 Conformance Requirements

The method with following header shall be compliant to its definition in the API.

6.2.4.5.1.1 Normal execution

- CRRN1: The method shall return the single system instance of the EnvelopeHandler class.
- CRRN2: The EnvelopeHandler is a Temporary JCRE Entry Point Object (see Javacard 2.1 Runtime Environment (JCRE) Specification [12])

6.2.4.5.1.2 Parameters error

No requirements.

6.2.4.5.1.3 Context errors

• CRRC1: The method shall thrown ToolkitException (HANDLER_NOT_AVAILABLE) if the handler is busy.

6.2.4.5.2 Test suite files

Test Script: API_2_ENH_GTHD_1.scr
Test Applet: API_2_ENH_GTHD_1.java
Load Script: API_2_ENH_GTHD_1.ldr
Cleanup Script: API_2_ENH_GTHD_1.clr
Parameter File: API_2_ENH_GTHD_1.par

6.2.4.5.3 Test Procedure

ld	Description	API Expectation	APDU Expectation
1	getTheHandler twice	The returned objects shall be the	
		same	
2	Verify that getTheHandler returns an	The reference returned shall be an	
	EnvelopeHandler	EnvelopeHandler (check cast)	
	GetTheHandler		
3	Verify the returned value is not null	The reference returned shall not be	
	GetTheHandler	null.	

6.2.4.5.4 Test Coverage

CRR number	Test case number	
N1	1, 2, 3	
N2	To be checked in Framework tests and insert here cross reference	
C1	To be checked in Framework tests and insert here cross reference	

6.2.4.6 Method getTPUDLOffset

Test Area Reference: API_2_ENH_GTPO

6.2.4.6.1 Conformance Requirement:

The method with following header shall be compliant to its definition in the API.

6.2.4.6.1.1 Normal execution

- CRRN1: The method shall return the TPUDL offset in a SMS TPDU TLV.
- CRRN2: The offset is from the first SMS TPDU TLV.
- CRRN3: The method can be used if the event is EVENT_FORMATTED_SMS_PP_ENV.
- CRRN4: The method can be used if the event is EVENT_FORMATTED_SMS_PP_UPD.
- CRRN5: The method can be used if the event is EVENT_UNFORMATTED_SMS_PP_ENV.
- CRRN6: The method can be used if the event is EVENT_UNFORMATTED_SMS_PP_UPD.
- CRRN7: If the method is successful, the selected TLV should be the SMS TPDU TLV.

6.2.4.6.1.2 Parameters error

No requirements.

6.2.4.6.1.3 Context errors

- CRRC1: The method shall thrown ToolkitException (UNAVAILABLE_ELEMENT) in case of unavailable SMS TPDU TLV element.
- CRRC2: The method shall thrown ToolkitException (UNAVAILABLE_ELEMENT) if the TPUDL field does
 not exist.

6.2.4.6.2 Test suite files

Specific triggering:

- FORMATTED SMS PP UPD.
- UNFORMATTED SMS PP UPD.
- UNFORMATTED SMS PP ENV.
- UNFORMATTED SMS CB.

Test Script: API_2_ENH_GTPO_1.scr

Test Applet: API_2_ENH_GTPO_1.java

Load Script: API_2_ENH_GTPO_1.ldr

Cleanup Script: API_2_ENH_GTPO_1.clr

Parameter File: API_2_ENH_GTPO_1.par

6.2.4.6.3 Test procedure

ld	Description	API Expectation	APDU Expectation
	FORMATTED SMS PP ENV triggering		
1	Test with TP-OA length of 2	Returns 0x0D	
2	Test with TP-OA length of 6	Returns 0x0F	
3	Test with TP-OA length of 12	Returns 0x12	
4	Send a SMS PP with 2 TPDU TLV and inside two different UDL offsets	Returns 0x10 (the first offset)	
5	Same test as 1, but with a concatenated SMS (2 Short Messages and maximum Secured Data Length = 0x00FA)	Returns 0x0D	
6	Verify after call of the method the current TLV is the TPDU TLV: findTLV device identities, getTPUDLOffset and then getValueByte to verify that the current TLV is the TPDU TLV FORMATTED SMS PP UPD triggering	Returns 0x40	
7	Same test as 1 but with	Returns 0x0D	
	FORMATTED_SMS_PP_UPD		
	Same test as 2 but with FORMATTED_SMS_PP_UPD	Returns 0x0F	
9	Same test as 3 but with FORMATTED_SMS_PP_UPD	Returns 0x12	
	Same test as 4 but with FORMATTED_SMS_PP_UPD	Returns 0x10 (the first offset)	
11	Same test as 7, but with a concatenated SMS (2 Short Messages and maximum Secured Data Length = 0x00FA)	Returns 0x0D	
40	UNFORMATTED SMS PP UPD triggering	D-1	
	Same test as 1 but with UNFORMATTED_SMS_PP_UPD	Returns 0x0D	
	Same test as 2 but with UNFORMATTED_SMS_PP_UPD	Returns 0x0F	
14	Same test as 3 but with UNFORMATTED_SMS_PP_UPD	Returns 0x12	
	Same test as 4 but with UNFORMATTED_SMS_PP_UPD	Returns 0x12 (the first offset)	
16	Same test as 12, but with a concatenated SMS (2 Short Messages and maximum User Data Length = 0x010C)	Returns 0x0D	
47	UNFORMATTED SMS PP ENV triggering	Datum - 0.0D	
17	Same test as 1 but with UNFORMATTED_SMS_PP_ENV	Returns 0x0D	
18	Same test as 2 but with UNFORMATTED_SMS_PP_ENV	Returns 0x0F	
19	Same test as 3 but with UNFORMATTED_SMS_PP_ENV	Returns 0x12	
20	Same test as 4 but with UNFORMATTED_SMS_PP_ENV	Returns 0x10 (the first offset)	
21	Same test as 17, but with a concatenated SMS (2 Short Messages and maximum User Data Length = 0x010C) SMS CB triggering	Returns 0x0D	
22	Send an envelope SMS CB, getTPUDLOffset	ToolkitException	
	Sing an onvelope one ob, goth obtained	UNAVAILABLE_ELEMENT	

6.2.4.6.4 Test Coverage

CRR number	Test case number
N1	1 to 21.
N2	4, 10, 15, 20.
N3	1, 2, 3, 4, 5, 6
N4	7, 8, 9, 10, 11
N5	12, 13, 14, 15, 16
N6	17, 18, 19, 20, 21
N7	6
C1	22
C2	Not applicable

6.2.4.7 Method getLength

Test Area Reference: API_2_ENH_GLEN

6.2.4.7.1 Conformance Requirement

The method with following header shall be compliant to its definition in the API.

public short getLength()

throws ToolkitException

6.2.4.7.1.1 Normal execution

• CRRN1: returns the length in bytes of the TLV list.

6.2.4.7.1.2 Parameter Error

No requirements.

6.2.4.7.1.3 Context errors

• CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException.HANDLER_NOT_AVAILABLE.

6.2.4.7.2 Test Suite files

Specific triggering: None

Test Script: API_2_ENH_GLEN_1.scr

Test Applet: API_2_ENH_GLEN_1.java

Load Script: API_2_ENH_GLEN_1.ldr

Cleanup Script: API_2_ENH_GLEN_1.clr

Parameter File: API_2_ENH_GLEN_1.par

6.2.4.7.3 Test Procedure

ld	Description	API Expectation	APDU Expectation
1	Send an envelope SMS PP with BER length of 0x31	Result of getLength() is 0x0031	
2	Send an envelope SMS PP with BER length of 0x7F	Result of getLength() is 0x007Fh	
3	Send an envelope SMS PP with BER length of 81 80	Result of getLength() is 0x0080h	
4	Send an envelope SMS PP with BER length of 81 FC	Result of getLength() is 0x00FCh	
	(maximum length for a single SMS)		
5	Send formatted SMS with BER length of 0x00FF,	Result of getLength() is 0x00FFh	
	using 2 concatenated SMS		

	Send formatted SMS with BER length of 0x0100, using 2 concatenated SMS	Result of getLength() is 0x0100h	
7	Send formatted SMS with maximum user data length	Result of getLength() is 0x012Fh	
	(0x10D) (BER length:0x012F), using 2 concatenated		
	SMS		

6.2.4.7.4 Test Coverage

CRR number	Test case number
N1	1, 2, 3, 4, 5, 6, 7
C1	Does not apply for
	EnvelopeHandler

6.2.4.8 Method copy

Test Area Reference: API_2_ENH_COPY_BSS

6.2.4.8.1 Conformance Requirement

The method with following header shall be compliant to its definition in the API.

6.2.4.8.1.1 Normal execution

- CRRN1: copies the simple TLV list contained in the handler to the destination byte array.
- CRRN2: returns dstOffset + dstLength.

6.2.4.8.1.2 Parameter errors

- CRRP1: if dstBuffer is null a NullPointerException is thrown.
- CRRP2: if dstOffset or dstLength or both would cause access outside array bounds, or if dstLength is negative, an ArrayIndexOutOfBoundsException is thrown.
- CRRP3: if dstLength is grater than the length of the simple TLV List, an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException. OUT_OF_TLV_BOUNDARIES.

6.2.4.8.1.3 Context errors

• CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.

6.2.4.8.2 Test Suite files

Specific triggering: None

Test Script: API_2_ENH_COPY_BSS_1.scr
Test Applet: API_2_ENH_COPY_BSS_1.java
Load Script: API_2_ENH_COPY_BSS_1.ldr
Cleanup Script: API_2_ENH_COPY_BSS_1.clr
Parameter File: API_2_ENH_COPY_BSS_1.par

6.2.4.8.3 Test Procedure

ld	Description	API Expectation	APDU Expectation
1	NULL as parameter to dstBuffer	NullPointerException is thrown	
		F	
2	dstOffset ≥ dstBuffer.length	ArrayIndexOutOfBoundsException is	
	dstBuffer.length = 5	thrown	
	dstOffset = 5		
_	dstLength = 1	A manufactor of David de Everantian in	
3	<pre>dstOffset < 0 dstBuffer.length = 5</pre>	ArrayIndexOutOfBoundsException is thrown	
	dstOffset = -1	HIIOWII	
	dstLength = 1		
4	dstLength > dstBuffer.length	ArrayIndexOutOfBoundsException is	
	dstBuffer.length = 5	thrown	
	dstOffset = 0		
5	dstLength = 6	ArrayladayOutOfDayadaEyaantian is	
Э	DstOffset + dstLength > dstBuffer.length DstBuffer.length = 5	ArrayIndexOutOfBoundsException is thrown	
	DstOffset = 3	unown	
	DstLength = 3		
6	dstLength < 0	ArrayIndexOutOfBoundsException is	_
	dstBuffer.length = 5	thrown	
	dstOffset = 0 dstLength = -1		
7	DstLength > length of the simple TLV list	ToolkitException.OUT_OF_TLV_BO	
′	DstBuffer.length = 48	UNDARIES is thrown	
	DstOffset = 0	ONDANIES IS UITOWIT	
	DstLength = 48		
8	Successful call, dstBuffer is the whole buffer	Result of copy() is 0X0047	
	DstBuffer.length = 47		
	DstOffset = 0 DstLength = 47		
9	Compare the buffer	Result of arrayCompare() is 0	
10	Successful call, dstBuffer is part of a buffer	Result of copy() is 0X0032	
10	DstBuffer.length = 50	result of copy() is oxocoz	
	dstOffset = 3		
	dstLength = 47		
11	Compare the whole buffer	Result of arrayCompare() is 0	
12	Successful call, dstBuffer is part of a buffer	Result of copy() is 0X0009	
	<pre>dstBuffer.length = 15 dstOffset = 3</pre>		
	dstLength = 6		
13	Compare the whole buffer	Result of arrayCompare() is 0	
	Successful call, dstBuffer is part of a buffer	Result of copy() is 0X0104	
	dstBuffer.length = 260		
	dstOffset = 257		
15	dstLength = 3	Popult of arrayCompara() is 0	
15 16	Compare the whole buffer Successful call, copy with length =0	Result of arrayCompare() is 0 Result of copy() is 0x104	
10	dstBuffer.length = 260	Result of copy() is 0x104	
	dstOffset = 260		
	dstLength = 0		
	Send a Formatted SMS PP with the maximum		
	user data length = 0x010D, using 2		
	concatenated envelopes		
17	Successful call, copy with length =299	Result of copy() is 0x12B	
	dstBuffer.length = 299		
	dstOffset = 0 dstLength = 299		
	นอเนยเเชนา = 299		

6.2.4.8.4 Test Coverage

CRR number Test case number	
N1	9, 11, 13, 15
N2	8, 10, 12, 14, 16, 17
P1	1
P2	2, 3, 4, 5, 6
P3	7
C1	Does not apply for EnvelopeHandler

6.2.4.9 Method findTLV

Test Area Reference: API_2_ENH_FINDBB

6.2.4.9.1 Conformance Requirement

The method with following header shall be compliant to its definition in the API.

public byte findTLV(byte tag, byte occurrence) throws ToolkitException

6.2.4.9.1.1 Normal execution

Looks for the indicated occurrence of a TLV element from the beginning of the TLV list (handler buffer):

- CRRN1: the method is successful if the required occurrence exists then the corresponding TLV becomes current.
- CRRN2: if the method is successful then it returns TLV_FOUND_CR_SET when Comprehension Required flag is set.
- CRRN3: if the method is successful then it returns TLV_FOUND_CR_NOT_SET when Comprehension Required flag is not set.
- CRRN4: if the required occurrence of the TLV element does not exist, the current TLV is no longer defined and TLV_NOT_FOUND is returned.
- CRRN5: The search method is comprehension required flag independent.

6.2.4.9.1.2 Parameter errors

• CRRP1: if an input parameter is not valid (e.g. occurrence = 0) an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException BAD_INPUT_PARAMETER.

6.2.4.9.1.3 Context errors

CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.

6.2.4.9.2 Test Suite files

Specific triggering: None

Cleanup Script:

Test Script: API_2_ENH_FINDBB_1.scr Test Applet: API_2_ENH_FINDBB_1.java Load Script: API_2_ENH_FINDBB_1.ldr

API_2_ENH_FINDBB_1.clr Parameter File: API_2_ENH_FINDBB_1.par

6.2.4.9.3 **Test Procedure**

ld	Description	API Expectation	APDU Expectation
	Trig the applet with SMS PP including one more		
	tag 02 and one TAG 04		
1	Invalid input parameter	ToolkitException.BAD_INPUT_PA	
	Occurrence = 0	RAMETER is thrown	
2	Search 1st TLV	Result is TLV_FOUND_CR_SET	
	Tag = 02h		

ld	Description	API Expectation	APDU Expectation
	Occurrence = 1	·	•
3	Call the getValueLength() method	Result is 0x02	
4	Search 2nd TLV	Result is TLV_FOUND_CR_SET	
	Tag = 06h		
	Occurrence = 1		
5	Call the getValueLength() method	Result is 0x05h	
6	Select a TLV (tag 02h)		
	Search a wrong tag	Result is TLV_NOT_FOUND	
	Tag = 03h		
	Occurrence = 1		
7	Call the getValueLength() method	ToolkitException.UNAVAILABLE_	
		ELEMENT is thrown.	
8	Search a tag with wrong occurrence	Result is TLV_NOT_FOUND	
	Tag = 02h		
_	Occurrence = 3	T 11:15 C 11N1AV/AH ADLE	
9	Call the getValueLength() method	ToolkitException.UNAVAILABLE_	
40	Connels the TLV	ELEMENT is thrown.	
10	Search the TLV	Result is	
	Tag = 02h Occurrence = 2	TLV_FOUND_CR_NOT_SET	
11	Search the TLV	Result is	
	Tag = 04h	TLV FOUND CR NOT SET	
	Occurrence = 1	TEV_TOOND_OR_NOT_OFT	
12	Search tag 86h	Result is TLV_FOUND_CR_SET	
	Tag = 86h		
	Occurrence = 1		
13	Search tag 84h	Result is	
	Tag = 84h	TLV_FOUND_CR_NOT_SET	
	Occurrence = 1		

6.2.4.9.4 Test Coverage

CRR number	Test case number	
N1	3, 5	
N2	2, 4	
N3	10, 11	
N4	6, 7, 8, 9	
N5	12, 13	
P1	1	
C1	Does not apply for EnvelopeHandler	

6.2.4.10 Method getValueLength

Test Area Reference: API_2_ENH_GVLE

6.2.4.10.1 Conformance Requirement

The method with following header shall be compliant to its definition in the API.

6.2.4.10.1.1 Normal execution

• CRRN1: gets and returns the binary length of the value field for the last TLV element which has been found in the handler.

6.2.4.10.1.2 Parameter errors

No requirements.

6.2.4.10.1.3 Context errors

- CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.
- CRRC2: in case of unavailable TLV element an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException UNAVAILABLE_ELEMENT.

6.2.4.10.2 Test Suite files

Specific triggering: None

Test Script: API_2_ENH_GVLE_1.scr

Test Applet: API_2_ENH_GVLE_1.java

Load Script: API_2_ENH_GVLE_1.ldr

Cleanup Script: API_2_ENH_GVLE_1.clr

Parameter File: API_2_ENH_GVLE_1.par

6.2.4.10.3 Test Procedure

ld	Description	API Expectation	APDU Expectation
	Fill the SMS PP with TLV: Tag 33, Length C8		
1	getValueLength()	ToolkitException.UNAVAILABLE_	
		ELEMENT is thrown	
2	Search TLV 02h		
	getValueLength()	Result is 0X0002	
3	Search TLV 0Bh		
	getValueLength()	Result is 0X0024	
4	Search TLV 33h		
	getValueLength()	Result is 0X00C8	
	Send Formatted SMS PP with the maximum		
	user data length = 0x010D, using 2		
	concatenated envelopes		
5	Search SMS TPDU TAG		·
	getValueLength()	Result is 0X0120	

6.2.4.10.4 Test Coverage

CRR number	Test case number	
N1	2, 3, 4, 5	
C1	Does not apply for EnvelopeHandler	
C2	1	

6.2.4.11 Method getValueByte

Test Area Reference: API_2_ENH_GVBYS

6.2.4.11.1 Conformance Requirement

The method with following header shall be compliant to its definition in the API.

 $\label{eq:public_public} \mbox{public byte getValueByte(short valueOffset)} \\ \mbox{throws ToolkitException}$

6.2.4.11.1.1 Normal execution

• CRRN1: Gets a byte from the last TLV element which has been found in the handler and returns its value (1 byte).

6.2.4.11.1.2 Parameter errors

• CRRP1: if valueOffset is out of the current TLV an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException OUT_OF_TLV_BOUNDARIES.

6.2.4.11.1.3 Context errors

- CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.
- CRRC2: in case of unavailable TLV element an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException UNAVAILABLE_ELEMENT.

6.2.4.11.2 Test Suite files

Specific triggering: None

Test Script: API_2_ENH_GVBYS_1.scr

Test Applet: API_2_ENH_GVBYS_1.java

Load Script: API_2_ENH_GVBYS_1.dr

Cleanup Script: API_2_ENH_GVBYS_1.clr

Parameter File: API_2_ENH_GVBYS_1.par

6.2.4.11.3 Test Procedure

ld	Description	API Expectation	APDU Expectation
	Fill the SMS PP with TLV: Tag 33, Length C8		-
	Value 01 02		
1	getValueByte(0)	ToolkitException.UNAVAILABLE_	
		ELEMENT is thrown	
2	Search TLV 02h		
	getValueByte(2)	ToolkitException.OUT_OF_TLV_	
		BOUNDARIES is thrown	
3	Search TLV 02h		
	getValueByte(1)	Result is 0x81	
4	Search TLV 02h (Device Identities TLV)		
	getValueByte(0)	Result is 83h (Source)	
5	Search TLV 33h		
	getValueByte(7E)	Result is 0x7F	
6	Search TLV 33h		
	getValueByte(80)	Result is 0x81	
7	getValueByte(7F)	Result is 0x80	
8	Search TLV B3h		
	getValueByte(C7)	Result is 0xC8	
	Send Formatted SMS PP with the maximum		
	user data length = 0x010D, using 2		
	concatenated envelopes		
9	Search SMS TPDU TAG		
	getValueByte(0x011F)	Result is 0xFA	

6.2.4.11.4 Test Coverage

CRR number	Test case number
N1	3, 4, 5, 6, 7, 8, 9
P1	2
C1	Does not apply for EnvelopeHandler
C2	1

6.2.4.12 Method copyValue

Test Area Reference: API_2_ENH_CPYVS_BSS

6.2.4.12.1 Conformance Requirement

The method with following header shall be compliant with its definition in the API.

6.2.4.12.1.1 Normal execution

- CRRN1: copies a part of the last TLV element which has been found, into a destination. buffer.
- CRRN2: returns dstOffset + dstLength.

6.2.4.12.1.2 Parameter errors

- CRRP1: if dstBuffer is null NullPointerException is thrown.
- CRRP2: if dstOffset or dstLength or both would cause access outside array bounds, or if dstLength is negative ArrayIndexOutOfBoundsException is thrown.
- CRRP3: if valueOffset, dstLength or both are out of the current TLV an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException OUT OF TLV BOUNDARIES.

6.2.4.12.1.3 Context errors

- CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER NOT AVAILABLE.
- CRRC2: in case of unavailable TLV element an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException UNAVAILABLE_ELEMENT.

6.2.4.12.2 Test Suite files

Specific triggering: None

Test Script: API_2_ENH_CPYVS_BSS_1.scr

Test Applet: API_2_ENH_CPYVS_BSS_1.java

Load Script: API_2_ENH_CPYVS_BSS_1.ldr

Cleanup Script: API_2_ENH_CPYVS_BSS_1.clr

Parameter File: API_2_ENH_CPYVS_BSS_1.par

6.2.4.12.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Search TLV 02h		
	copyValue() with a null dstBuffer	NullPointerException is thrown	
2	Search TLV 0Bh		
	dstOffset ≥ dstBuffer.length	ArrayIndexOutOfBoundsExceptio	
	dstBuffer.length = 5	n is thrown	
	dstOffset = 5		
	dstLength = 1		
3	dstOffset < 0	ArrayIndexOutOfBoundsExceptio	

ld	Description	API Expectation	APDU Expectation
	dstBuffer.length = 5	n is thrown	-
	dstOffset = -1		
_	dstLength = 1	A 1 1 0 10'D 1 5	
4	<pre>dstLength >dstBuffer.length dstBuffer.length = 5</pre>	ArrayIndexOutOfBoundsExceptio	
	dstOffset = 0	n is thrown	
	dstLength = 6		
5	dstOffset + dstLength >dstBuffer.length	ArrayIndexOutOfBoundsExceptio	
	dstBuffer.length = 5	n is thrown	
	dstOffset = 3		
6	dstLength = 3 dstLength < 0	ArrayIndexOutOfBoundsExceptio	
0	dstBuffer.length = 5	n is thrown	
	dstOffset = 0	II IS UIIOWII	
	dstLength = -1		
7	Search TLV 06h		
	valueOffset ≥ TLV Length	ToolkitException.OUT_OF_TLV_	
	valueOffset = 6	BOUNDARIES is thrown	
	<pre>dstBuffer.length = 15 dstOffset = 0</pre>		
	dstLength = 1		
8	valueOffset < 0	ToolkitException.OUT_OF_TLV_	
	valueOffset = -1	BOUNDARIES is thrown	
	<pre>dstBuffer.length = 15 dstOffset = 0</pre>		
	dstLength = 1		
9	dstLength > TLV length	ToolkitException.OUT_OF_TLV_	
	valueOffset = 0	BOUNDARIES is thrown	
	dstBuffer.length = 15		
	dstOffset = 0		
40	dstLength = 7	Tablithan OUT OF TIV	
10	valueOffset + dstLength > TLV length valueOffset = 2	ToolkitException.OUT_OF_TLV_ BOUNDARIES is thrown	
	dstBuffer.length = 15	BOONDAINES IS UNOWIT	
	dstOffset = 0		
	dstLength = 5		
11	Search TLV 01h	THitFire anti-in LINIAY/AH ADI F	
	copyvalue()	ToolkitException.UNAVAILABLE_ ELEMENT is thrown on the	
		copyValue() method	
12	Search TLV 06h	copy value() method	
12	Successful call	Result of copyValue() is 0x0006	
	valueOffset = 0	result of copy value() is exceed	
	dstBuffer.length = 6		
	dstOffset = 0		
13	dstLength = 6 Compare buffer	Result is 00h	
13	buffer = 81 11 22 33 44 F5	Result is our	
14	initialise dstBuffer		
	dstBuffer = 55 55 55		
	Successful call	Result of copyValue() is 0x0007	
	<pre>valueOffset = 1 dstBuffer.length = 20</pre>		
	dstOffset = 3		
	dstLength = 4		
15	Compare buffer	Result is 00h	
	buffer =		
	55 55 55 11 22 33 44 55 55 55		
	55 55 55 55 55		
	55 55 55 55		
16	Successful call, copy with length =0	Result of copyValue() is 20	
	dstBuffer.length = 20		
	dstOffset = 20 dstLength = 0		
	Send Formatted SMS with the maximum user		
	data length = 0x010D, using 2 concatenated		
	envelopes		
	-	1	

ld	Description	API Expectation	APDU Expectation
17	Search SMS TPDU TAG		-
	Successful call	Result of copyValue() is 0x010D	
	valueOffset = 0x11	·	
	dstBuffer.length = 0x010D		
	dstOffset = 0		
	dstLength = 0x010D		
18	Compare buffer	Result is 00h	
	buffer = 0348 header and secured data (01		
	FA)		
19	Initialise dstBuffer		
	dstBuffer = 55 55 55		
	Successful call	Result of copyValue() is 0x010D	
	valueOffset = 0x0111		
	dstBuffer.length = 0x010D		
	dstOffset = 0x0100		
	dstLength = 0x000D		
20	Compare buffer	Result is 00h	
	buffer =		
	55 55 55 55 55 55 55		
	55 55 EE EF F0 F1 F2 F3 F4 F5 F6 F7 F8 F9		
	FA		

6.2.4.12.4 Test Coverage

CRR number	Test case number
N1	13, 15, 18, 20
N2	12, 14, 16, 17, 19
P1	1
P2	2, 3, 4, 5, 6
P3	7, 8, 9, 10
C1	Does not apply for EnvelopeHandler
C2	11

6.2.4.13 Method compareValue

Test Area Reference: API_2_ENH_CPRVS_BSS

6.2.4.13.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

6.2.4.13.1.1 Normal execution

Compares the last found TLV element with a buffer:

- CRRN1: returns 0 if identical.
- CRRN2: returns -1 if the first miscomparing byte in simple TLV List is less than that in compareBuffer.
- CRRN3: returns 1 if the first miscomparing byte in simple TLV List is greater than that in compareBuffer.

6.2.4.13.1.2 Parameter errors

• CRRP1: if compareBuffer is null NullPointerException shall be thrown.

- CRRP2: if compareOffset or compareLength or both would cause access outside array bounds, or if compareLength is negative ArrayIndexOutOfBoundsException shall be thrown.
- CRRP3: if valueOffset, dstLength or both are out of the current TLV an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException OUT_OF_TLV_BOUNDARIES.

6.2.4.13.1.3 Context errors

- CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.
- CRRC2: in case of unavailable TLV element an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException UNAVAILABLE_ELEMENT.

6.2.4.13.2 Test Suite files

Specific triggering: None

Test Script: API_2_ENH_CPRVS_BSS_1.scr

Test Applet: API_2_ENH_CPRVS_BSS_1.java

Load Script: API_2_ENH_CPRVS_BSS_1.ldr

Cleanup Script: API_2_ENH_CPRVS_BSS_1.clr

Parameter File: API_2_ENH_CPRVS_BSS_1.par

6.2.4.13.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Search TLV 02h	711	7.1. DO Expositation
	compareValue() with a null compareBuffer	NullPointerException is thrown	
2	Search TLV 0Bh		
	<pre>compareOffset ≥ compareBuffer.length compareBuffer.length = 5 compareOffset = 5 compareLength = 1</pre>	ArrayIndexOutOfBoundsExceptio n is thrown	
3	<pre>compareOffset < 0 compareBuffer.length = 5 compareOffset = -1 compareLength = 1</pre>	ArrayIndexOutOfBoundsExceptio n is thrown	
4	<pre>compareLength >compareBuffer.length compareBuffer.length = 5 compareOffset = 0 compareLength = 6</pre>	ArrayIndexOutOfBoundsException is thrown	
5	<pre>compareOffset + compareLength</pre>	ArrayIndexOutOfBoundsExceptio n is thrown	
6	<pre>compareLength < 0 compareBuffer.length = 5 compareOffset = 0 compareLength = -1</pre>	ArrayIndexOutOfBoundsExceptio n is thrown	
7	Search TLV 06h		
	<pre>valueOffset ≥ TLV Length valueOffset = 6 compareBuffer.length = 15 compareOffset = 0 compareLength = 1</pre>	ToolkitException.OUT_OF_TLV_ BOUNDARIES is thrown	
8	<pre>valueOffset < 0 valueOffset = -1 compareBuffer.length = 15 compareOffset = 0 compareLength = 1</pre>	ToolkitException.OUT_OF_TLV_ BOUNDARIES is thrown	
9	compareLength > TLV length valueOffset = 0	ToolkitException.OUT_OF_TLV_ BOUNDARIES is thrown	

ld	Description	API Expectation	APDU Expectation
	compareBuffer.length = 15	·	•
	compareOffset = 0		
10	compareLength = 7 valueOffset + compareLength > TLV length	ToolkitException.OUT_OF_TLV_	
10	valueOffset = 2	BOUNDARIES is thrown	
	compareBuffer.length = 15		
	compareOffset = 0		
11	compareLength = 5 Search TLV 01h	Result is TLV_NOT_FOUND	
	compareValue()	ToolkitException.UNAVAILABLE_	
		ELEMENT is thrown	
12	Search TLV 06h		
	Initialise compareBuffer		
	compareBuffer = 81 11 22 33 44 F5		
	Compare buffers	Result is 00h	
	valueOffset = 0		
	<pre>compareOffset = 0 compareLength = 6</pre>		
13	Initialise compareBuffer		
	compareBuffer =		
	7F 11 22 33 44 F5		
14	Compare buffers with same parameters Initialise compareBuffer	Result is -1	
14	compareBuffer =		
	83 11 22 33 44 F5		
	Compare buffers with same parameters	Result is -1	
15	Initialise compareBuffer compareBuffer =		
	55 55 55 81 11 22 33 44 F5 55 55 55 55 55		
	Compare buffers	Result is 00h	
	valueOffset = 1		
	compareOffset = 4		
16	compareLength = 5 Initialise compareBuffer		
	compareBuffer =		
	55 55 55 81 10 22 33 44 F5		
	55 55 55 55 55 Compare buffers with same parameters	Result is +1	
17	Initialise compareBuffer	Tresult is 11	
	compareBuffer =		
	55 55 55 81 12 22 33 44 F5		
	55 55 55 55 55 Compare buffers with same parameters	Result is -1	
	,		
18	Successful call, compareValue with length =0	Result of compareValue() is 0	
	CompareBuffer.length = 15 CompareOffset = 15		
	CompareLength = 0		
	Send Formatted SMS PP with the maximum		
	user data length = 0x010D, using 2		
-	concatenated envelopes Search SMS TPDU TAG		
<u> </u>	Initialise compareBuffer		
	compareBuffer = 0348 header and formatted		
1-	data(01 02 FA)		
19	Compare buffers	Result is 00h	
	<pre>valueOffset = 0x11 compareOffset = 0</pre>		
	compareLength = 0x010D		
	compareBufferLength = 0x010D	Deput is 00h	
20	Compare buffers valueOffset = 0x0111	Result is 00h	
	compareOffset = 0x0100		
	compareLength = 0x000D		
	compareBufferLength = 0x010D		

6.2.4.13.4 Test Coverage

CRR number	Test case number
N1	12, 15, 19, 20
N2	13, 16, 18
N3	14, 17
P1	1
P2	2, 3, 4, 5, 6
P3	7, 8, 9, 10
C1	Does not apply for EnvelopeHandler
C2	11

6.2.4.14 Method findAndCopyValue(byte tag, byte[] dstBuffer, short dstOffset)

Test Area Reference: API_2_ENH_FACYB_BS

6.2.4.14.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

6.2.4.14.1.1 Normal execution

- CRRN1: looks for the first occurrence of a TLV element from the beginning of a TLV list and copy its value into a destination buffer.
- CRRN2: if no TLV element is found, the UNAVAILABLE_ELEMENT exception is thrown and the current TLV is no longer defined.
- CRRN3: if the method is successful then the corresponding TLV becomes current and dstOffset + length of the copied value is returned.
- CRRN4: The search method is comprehension required flag independent.

6.2.4.14.1.2 Parameter errors

- CRRP1: if dstBuffer is null NullPointerException shall be thrown.
- CRRP2: if dstOffset would cause access outside array bounds ArrayIndexOutOfBoundsException shall be thrown.

6.2.4.14.1.3 Context errors

• CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.

6.2.4.14.2 Test Suite files

Specific triggering: None

Test Script: API_2_ENH_FACYB_BS_1.scr
Test Applet: API_2_ENH_FACYB_BS_1.java
Load Script: API_2_ENH_FACYB_BS_1.ldr
Cleanup Script: API_2_ENH_FACYB_BS_1.clr
Parameter File: API_2_ENH_FACYB_BS_1.par

6.2.4.14.3 Test procedure

ld	Description	API Expectation	APDU Expectation
	Fill the SMS PP with TLV: Tag 02 Value 22 44		
	Tag 33, Length C4 Value 01 02	N. 115	
1	FindAndCopyValue() with a null dstBuffer	NullPointerException is thrown	
2	<pre>dstOffset ≥ dstBuffer.length tag = 06h dstBuffer.length = 06 dstOffset = 06</pre>	ArrayIndexOutOfBoundsException is thrown	
3	<pre>dstOffset < 0 dstBuffer.length = 06 dstOffset = -1</pre>	ArrayIndexOutOfBoundsException is thrown	
4	<pre>length > dstBuffer.length dstBuffer.length = 05 dstOffset = 0</pre>	ArrayIndexOutOfBoundsException is thrown	
5	DstOffset + length >dstBuffer.length DstBuffer.length = 06 DstOffset = 1	ArrayIndexOutOfBoundsException is thrown	
6	Select a TLV (tag 02h)		
	findAndCopyValue() tag = 03h	ToolkitException.UNAVAILABLE_ ELEMENT is thrown	
7	Call the getValueLength() method	ToolkitException.UNAVAILABLE_ ELEMENT is thrown.	
8	Successful call Tag = 06h DstBuffer.length = 06 DstOffset = 0	Result of findAndCopyValue () is 0006	
9	Compare buffer buffer = 81 11 22 33 44 F5	Result is 00h	
10	initialise dstBuffer dstBuffer = 55 55 55		
	Successful call dstBuffer.length = 12 dstOffset = 2	Result of findAndCopyValue () is 0008	
11	Compare buffer buffer = 55 55 81 11 22 33 44 F5 55 55 55 55	Result is 00h	
12	Successful call tag = 02h dstBuffer.length = 2 dstOffset = 0	Result of findAndCopyValue () is 0002	
13	Compare buffer buffer = 83 81	Result is 00h	
14	Successful call (with tag 82h) tag = 82h dstBuffer.length = 02 dstOffset = 0	Result of findAndCopyValue () is 0002	
15	Compare buffer buffer = 83 81	Result is 00h	
16	Successful call (with tag B3h) tag = B3h dstBuffer.length = C4 dstOffset = 0	Result of findAndCopyValue () is 00C4	

ld	Description	API Expectation	APDU Expectation
17	Compare buffer	Result is 00h	
	buffer = 01 02 C4		
	Send Formatted SMS PP with the maximum		
	user data length = 0x010D, using 2		
	concatenated envelopes		
18	Successful call (with SMS TPDU TAG)	Result of findAndCopyValue () is	
	tag = 0Bh	0x011E	
	dstBuffer.length = 0x011E		
	dstOffset = 0		
19	Compare buffer	Result is 00h	
	buffer = 0348 Header + secured data (01		
	02 FA)		
20	Successful call (with SMS TPDU TAG)	Result of findAndCopyValue () is	
	tag = 0Bh	0x021E	
	dstBuffer.length = 0x0220		
	dstOffset = 0x0100		
21	Compare buffer	Result is 00h	
	buffer = 0348 Header + secured data (01		
	02 FA)		

6.2.4.14.4 Test Coverage

CRR number	Test case number
N1	9, 11, 13
N2	6, 7
N3	8, 10, 12
N4	14, 15, 16, 17, 18, 19, 20, 21
P1	1
P2	2, 3, 4, 5
C1	Does not apply for EnvelopeHandler

6.2.4.15 Method findAndCopyValue(byte tag, byte occurrence, short valueOffset, byte[] dstBuffer, short dstOffset, short dstLength)

Test Area Reference: API_2_ENH_FACYBS_BSS

6.2.4.15.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

6.2.4.15.1.1 Normal execution

- CRRN1: looks for the indicated occurrence of a TLV element from the beginning of a TLV list and copy its value into a destination buffer.
- CRRN2: if no TLV element is found, the UNAVAILABLE_ELEMENT exception is thrown and the current TLV is no longer defined.
- CRRN3: if the method is successful then the corresponding TLV becomes current and dstOffset + dstLength is returned.
- CRRN4: The search method is comprehension required flag independent.

6.2.4.15.1.2 Parameter errors

- CRRP1: if dstBuffer is null NullPointerException shall be thrown.
- CRRP2: if dstOffset or dstLength or both would cause access outside array bounds, or if dstLength is negative ArrayIndexOutOfBoundsException shall be thrown.
- CRRP3: if valueOffset, dstLength or both are out of the current TLV an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException OUT_OF_TLV_BOUNDARIES.

6.2.4.15.1.3 Context errors

• CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.

6.2.4.15.2 Test Suite files

Test Script: API_2_ENH_FACYBS_BSS_1.scr
Test Applet: API_2_ENH_FACYBS_BSS_1.java
Load Script: API_2_ENH_FACYBS_BSS_1.ldr
Cleanup Script: API_2_ENH_FACYBS_BSS_1.clr
Parameter File: API_2_ENH_FACYBS_BSS_1.par

6.2.4.15.3 Test procedure

ld	Description	API Expectation	APDU Expectation
	Fill the SMS PP with TLV: Tag 02 Value 22 44		
	Tag 33, Length C4 Value 01 02		
1	findAndCopyValue() with a null dstBuffer	NullPointerException is thrown	
2	dstOffset ≥ dstBuffer.length	ArrayIndexOutOfBoundsExceptio	
	tag = 06h, occurrence = 1	n is thrown	
	valueOffset = 0		
	<pre>dstBuffer.length = 5 dstOffset = 5</pre>		
	dstOffset = 5 dstLength = 1		
3	dstOffset < 0	ArrayIndexOutOfBoundsExceptio	
٦	dstBuffer.length = 5	n is thrown	
	dstOffset = -1	II IS UIIOWII	
	dstLength = 1		
4	dstLength >dstBuffer.length	ArrayIndexOutOfBoundsExceptio	
	dstBuffer.length = 5	n is thrown	
	dstOffset = 0		
	dstLength = 6		
5	dstOffset + dstLength >dstBuffer.length	ArrayIndexOutOfBoundsExceptio	
	<pre>dstBuffer.length = 5 dstOffset = 3</pre>	n is thrown	
	dstOffset = 3 dstLength = 3		
6	dstLength < 0	ArrayIndexOutOfBoundsExceptio	
0	dstBuffer.length = 5	n is thrown	
	dstOffset = 0	II IS UIIOWII	
	dstLength = -1		
7	valueOffset ≥ Value Length	ToolkitException.OUT_OF_TLV_	
	tag = 06h, occurrence = 1	BOUNDARIES is thrown	
	valueOffset = 6		
	dstBuffer.length = 15		
	dstOffset = 0 dstLength = 1		
8	valueOffset < 0	ToolkitException.OUT_OF_TLV_	
١	valueOffset = -1	BOUNDARIES is thrown	
	dstBuffer.length = 15	DOGINDAINES IS UIIOWII	
	dstOffset = 0		
	dstLength = 1		
9	dstLength > Value length	ToolkitException.OUT_OF_TLV_	
	valueOffset = 0	BOUNDARIES is thrown	

ld	Description	API Expectation	APDU Expectation
	dstBuffer.length = 15		
	dstOffset = 0		
4.0	dstLength = 7	T 11:25 01 01 T 05 T1 V	
10	<pre>valueOffset + dstLength > Text String length valueOffset = 2</pre>	ToolkitException.OUT_OF_TLV_ BOUNDARIES is thrown	
	dstBuffer.length = 15	BOUNDARIES IS INFOWN	
	dstOffset = 0		
L.,	dstLength = 5		
11	Select a TLV (tag 02h)	T HEE C INVALABLE	
	findAndCopyValue() tag = 06h	ToolkitException.UNAVAILABLE_ ELEMENT is thrown	
	occurrence = 2	ELEMENT IS UTOWN	
12	Call the getValueLength() method	ToolkitException.UNAVAILABLE_	
		ELEMENT is thrown.	
13	Successful call	Result of findAndCopyValue() is 6	
	<pre>tag = 06h, occurrence = 1 valueOffset = 0</pre>		
	dstBuffer.length = 06		
	dstOffset = 0		
	dstLength = 06		
14	Compare buffer	Result is 00h	
15	buffer = 81 11 22 33 44 F5 initialise dstBuffer		
13	dstBuffer = 55 55 55		
	Successful call	Result of findAndCopyValue () is	
	tag = 06h, occurrence = 1	0007	
	valueOffset = 2 dstBuffer.length = 12		
	dstOffset = 3		
	dstLength = 04		
16	Compare buffer	Result is 00h	
	buffer =		
17	55 55 55 22 33 44 F5 55 55 55 55 55 Successful call	Result of findAndCopyValue() is	
' '	tag = 02h, occurrence = 1	0002	
	valueOffset = 0	0002	
	dstBuffer.length = 12		
	dstOffset = 0 dstLength = 2		
18	Compare buffer	Result is 00h	
	buffer = 83 81 55 55		
19	Successful call	Result of findAndCopyValue() is	
	<pre>tag = 02h, occurrence = 2 valueOffset = 0</pre>	0002	
	dstBuffer.length = 12		
	dstOffset = 0		
	dstLength = 2		
20	Compare buffer	Result is 00h	
21	buffer = 22 44 55 55 Successful call (with tag 82h)	Result of findAndCopyValue () is	
- '	tag = 82h	0002	
	occurrence = 1		
	<pre>valueOffset = 0 dstBuffer.length = 12</pre>		
	dstOffset = 0		
	dstLength = 02		
22	Compare buffer	Result is 00h	
23	buffer = 83 81 55 55 Successful call (with tag 82h)	Result of findAndCopyValue () is	
23	tag = 82h	10002	
	occurrence = 2		
	valueOffset = 0		
	<pre>dstBuffer.length = 12 dstOffset = 0</pre>		
L	dstLength = 02		
24	Compare buffer	Result is 00h	
	Buffer = 22 44 55 55		
25	Successful call, findAndCopyValue with length	Result of findAndCopyValue () is	
	=0 DstBuffer.length = 12	12	
	dstOffset = 12		
	dstLength = 0		
	Send Formatted SMS PP with the maximum		

ld	Description	API Expectation	APDU Expectation
	user data length = 0x010D, using 2		
	concatenated envelopes		
26	Successful call	Result of findAndCopyValue() is	
	tag = 0Bh, occurrence = 1	0x010D	
	valueOffset = 0x11		
	dstBuffer.length = 0x010D		
	dstOffset = 0		
	dstLength = 0x010D		
27	Compare buffer	Result is 00h	
	buffer = 0348 Header + secured data (01		
	02 FA)		
28	initialise dstBuffer		
	dstBuffer = 55 55 55		
	Successful call	Result of findAndCopyValue () is	
	tag = 0Bh, occurrence = 1	0x010D	
	valueOffset = 0x0111		
	dstBuffer.length = 0x010D		
	dstOffset = 0x0100		
	dstLength = 0x0D		
29	Compare buffer	Result is 00h	
	buffer =		
	55 55 55 55 EE EF F0 F1 F2 F3 F4 F5 F6		
	F7 F8 F9 FA		

6.2.4.15.4 Test Coverage

CRR number	Test case number
N1	14, 15, 17, 19, 20
N2	11, 12
N3	13, 15, 17, 19, 25
N4	21, 22, 23, 24, 26, 27, 28,29
P1	1
P2	2, 3, 4, 5, 6
P3	7, 8, 9, 10
C1	Does not apply for EnvelopeHandler

6.2.4.16 Method findAndCompareValue(byte tag, byte[] compareBuffer, short compareOffset)

Test Area Reference: API_2_ENH_FACRB_BS

6.2.4.16.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

6.2.4.16.1.1 Normal execution

Looks for the first occurrence of a TLV element from beginning of a TLV list and compare its value with a buffer:

- CRRN1: if no TLV element is found, the UNAVAILABLE_ELEMENT exception is thrown and the current TLV is no longer defined.
- CRRN2: if the method is successful then the corresponding TLV becomes current.
- CRRN3: if identical returns 0.
- CRRN4: if the first miscomparing byte in simple TLV is less than that in compareBuffer returns -1.

- CRRN5: if the first miscomparing byte in simple TLV is greater than that in compareBuffer returns 1.
- CRRN6: The search method is comprehension required flag independent.

6.2.4.16.1.2 Parameter errors

- CRRP1: if compareBuffer is null NullPointerException shall be thrown.
- CRRP2: if compareOffset would cause access outside array bounds ArrayIndexOutOfBoundsException shall be thrown.

6.2.4.16.1.3 Context errors

• CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.

6.2.4.16.2 Test Suite files

Test Script: API_2_ENH_FACRB_BS_1.scr

Test Applet: API_2_ENH_FACRB_BS_1.java

Load Script: API_2_ENH_FACRB_BS_1.ldr

Cleanup Script: API_2_ENH_FACRB_BS_1.clr

Parameter File: API_2_ENH_FACRB_BS_1.par

6.2.4.16.3 Test procedure

ld	Description	API Expectation	APDU Expectation
	Fill the SMS PP with TLV: Tag 02 Value 22 44 Tag 33, Length C4 Value 01 02		·
1	findAndCompareValue() with a null dstBuffer	NullPointerException is thrown	
2	<pre>compareOffset ≥ compareBuffer.length tag = 06h compareBuffer.length = 12 compareOffset = 12</pre>	ArrayIndexOutOfBoundsException is thrown	
3	<pre>compareOffset < 0 compareBuffer.length = 12 compareOffset = -1</pre>	ArrayIndexOutOfBoundsException is thrown	
4	<pre>length > compareBuffer.length compareBuffer.length = 05 compareOffset = 0</pre>	ArrayIndexOutOfBoundsException is thrown	
5	<pre>compareOffset + length > compareBuffer.length compareBuffer.length = 12 compareOffset = 7</pre>	ArrayIndexOutOfBoundsException is thrown	
6	Select a TLV (tag 02h)		
	findAndCompareValue() tag = 03h	ToolkitException.UNAVAILABLE_ ELEMENT is thrown	
7	Call the getValueLength() method	ToolkitException.UNAVAILABLE_ ELEMENT is thrown.	
8	Initialise compareBuffer compareBuffer = 81 11 22 33 44 F5		
	Compare buffers tag = 06h compareOffset = 0	Result is 00h	
9	Verify current TLV getValueLength()	Result is 06	
10	Initialise compareBuffer compareBuffer = 81 11 22 33 44 F4		
	Compare buffers with same parameters	Result is +1	
11	Initialise compareBuffer compareBuffer = 81 11 22 33 44 F6		
	Compare buffers with same parameters	Result is -1	
12	Initialise compareBuffer		

	compareBuffer =		
	55 55 81 11 22 33 44 F5 55 55 55 55		
	Compare buffers	Result is 00h	
	compareOffset = 2		
13	Initialise compareBuffer		
. •	compareBuffer =		
	55 55 83 81 55 55 55 55 55 55 55		
	Compare buffers	Result is 00h	
	compareOffset = 2	Tresuit is oon	
14	Initialise compareBuffer		
14	compareBuffer =		
	55 55 83 80 55 55 55 55 55 55 55		
		Decultie 14	
	Compare buffers	Result is +1	
4-	compareOffset = 2		
15	Initialise compareBuffer		
	compareBuffer =		
	55 55 83 82 55 55 55 55 55 55 55		
	Compare buffers	Result is -1	
	compareOffset = 2		
16	Initialise compareBuffer		
	compareBuffer =		
	83 81 55 55 55 55 55 55 55 55 55		
	Successful call (with tag 02h)	Result is 00h	
	tag = 02h		
	compareBuffer.length = 12		
	compareOffset = 0		
17	Initialise compareBuffer		
	CompareBuffer = 01 02 C4		
	Successful call (with tag B3h)	Result is 00h	
	Tag = B3h		
	CompareBuffer.length = C4		
	CompareOffset = 0		
	Send Unformatted SMS PP with the maximum		
	user data length = 0x010C, using 2		
	concatenated envelopes		
	Initialise compareBuffer		
	CompareBuffer = 0340 Header + user data		
	(00 01 02 FF 01 OC)		
18	Successful call (with SMS TPDU TAG)	Result is 00h	
10	Tag = 0Bh	Result is out	
	CompareBuffer.length = 0x011E		
	CompareOffset = 0		
-			
	Initialise compareBuffer		
	CompareBuffer = 55 55 55 CompareBuffer from offset 0x0100= 0340		
	-		
	Header + user data (00 01 02 FF 01 0C)		
10	Successful call (with SMS TPDU TAG)	Popult is 00b	
19	Tag = 0Bh	Result is 00h	
	CompareBuffer.length = 0x220		
	CompareOffset = 0x0100		
	COMPATEOTISEC - OXOTOO		

6.2.4.16.4 Test Coverage

CRR number	Test case number
N1	6,7
N2	9
N3	8, 12, 13, 18, 19
N4	10, 14
N5	11, 15
N6	16, 17
P1	1
P2	2, 3, 4, 5
C1	Does not apply for EnvelopeHandler

6.2.4.17 Method findAndCompareValue(byte tag, byte occurrence, short valueOffset, byte[] compareBuffer, short compareOffset, short compareLength)

Test Area Reference: API_2_ENH_FACRBBS_BSS

6.2.4.17.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

6.2.4.17.1.1 Normal execution

Looks for the indicated occurrence of a TLV element from the beginning of a TLV list and compare its value with a buffer:

- CRRN1: if no TLV element is found, the UNAVAILABLE_ELEMENT exception is thrown and the current TLV is no longer defined.
- CRRN2: if the method is successful then the corresponding TLV becomes current.
- CRRN3: if identical 0 is returned.
- CRRN4: if the first miscomparing byte in simple TLV is less than that in compareBuffer -1 is returned.
- CRRN5: if the first miscomparing byte in simple TLV is greater than that in compareBuffer 1 is returned
- CRRN6: The search method is comprehension required flag independent.

6.2.4.17.1.2 Parameter errors

- CRRP1: if compareBuffer is null NullPointerException shall be thrown.
- CRRP2: if compareOffset or compareLength or both would cause access outside array bounds, or if compareLength is negative ArrayIndexOutOfBoundsException shall be thrown.
- CRRP3: if valueOffset, compareLength or both are out of the current TLV an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException OUT_OF_TLV_BOUNDARIES.
- CRRP4: if an input parameter is not valid (e.g. occurence = 0) an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException BAD_INPUT_PARAMETER.

6.2.4.17.1.3 Context errors

• CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.

6.2.4.17.2 Test Suite files

Test Script: API_2_ENH_FACRBBS_BSS_1.scr
Test Applet: API_2_ENH_FACRBBS_BSS_1.java
Load Script: API_2_ENH_FACRBBS_BSS_1.ldr
Cleanup Script: API_2_ENH_FACRBBS_BSS_1.clr
Parameter File: API_2_ENH_FACRBBS_BSS_1.par

6.2.4.17.3 Test procedure

93

ld	Description	API Expectation	APDU Expectation
	Fill the SMS PP with TLV: Tag 02 Value 22 44	P	,
	Tag 33, Length C4 Value 01 02		
1	findAndCompareValue() with a null compareBuffer	NullPointerException is thrown	
2	compareOffset ≥ compareBuffer.length	ArrayIndexOutOfBoundsExceptio	
	tag = 06h, occurrence = 1	n is thrown	
	valueOffset = 0		
	<pre>compareBuffer.length = 6 compareOffset = 6</pre>		
	compareLength = 1		
3	compareOffset < 0	ArrayIndexOutOfBoundsExceptio	
	compareBuffer.length = 6	n is thrown	
	compareOffset = -1		
4	compareLength = 1 compareLength >compareBuffer.length	A manufactor of the control of the control	
4	compareLength > compareBuffer.length	ArrayIndexOutOfBoundsExceptio n is thrown	
	compareOffset = 0	II IS UIIOWII	
	compareLength = 6		
5	compareOffset + compareLength	ArrayIndexOutOfBoundsExceptio	
	>compareBuffer.length	n is thrown	
	compareBuffer.length = 5		
	<pre>compareOffset = 3 compareLength = 3</pre>		
6	compareLength < 0	ArrayIndexOutOfBoundsExceptio	
5	compareBuffer.length = 5	n is thrown	
	compareOffset = 0		
	compareLength = -1		
7	valueOffset ≥ Value Length	ToolkitException.OUT_OF_TLV_	
	<pre>tag = 06h, occurrence = 1 valueOffset = 6</pre>	BOUNDARIES is thrown	
	compareBuffer.length = 15		
	compareOffset = 0		
	compareLength = 1		
8	valueOffset < 0	ToolkitException.OUT_OF_TLV_	
	<pre>valueOffset = -1 compareBuffer.length = 15</pre>	BOUNDARIES is thrown	
	compareOffset = 0		
	compareLength = 1		
9	compareLength > Value length	ToolkitException.OUT_OF_TLV_	
	valueOffset = 0	BOUNDARIES is thrown	
	<pre>compareBuffer.length = 15 compareOffset = 0</pre>		
	compareLength = 7		
10	valueOffset + compareLength > Value length	ToolkitException.OUT_OF_TLV_	
	valueOffset = 2	BOUNDARIES is thrown	
	compareBuffer.length = 15		
	<pre>compareOffset = 0 compareLength = 5</pre>		
11	Invalid parameter	ToolkitException.BAD_INPUT_PA	
• •	occurrence = 0	RAMETER is thrown	
12	Select a TLV (tag 02h)	1 2 2	
	findAndCompareValue()	ToolkitException.UNAVAILABLE_	
	tag = 06h	ELEMENT is thrown	
40	occurrence = 2	Tablifficanting HAIAVAH ADI 5	
13	Call the getValueLength() method	ToolkitException.UNAVAILABLE_ ELEMENT is thrown.	
14	Initialise compareBuffer	ELCIVIEIN I 15 UIIOWII.	
14	compareBuffer = 81 11 22 33 44 F5		
	findAndCompareValue()	Result is 00h	
	tag = 06h, occurrence = 1		
	valueOffset = 0		
	<pre>compareOffset = 0 compareLength = 6</pre>		
15	Verify current TLV	Result is 0006	
IJ	getValueLength()	Tresuit is 0000	
16	Initialise compareBuffer		
	compareBuffer = 81 11 22 33 44 F4		
	Compare buffers with same parameters	Result is +1	
17	Initialise compareBuffer		
17	compareBuffer = 81 11 22 33 44 F6	Deput is 4	
17	Compare Buffer = 81 11 22 33 44 F6 Compare buffers with same parameters	Result is -1	

CompareBuffer = S5 55 52 23 34 44 P5 55 55 55	ld	Description	API Expectation	APDU Expectation
Compare buffers 2 2 2 2 2 2 2 2 2		compareBuffer =	·	•
valueOffset = 2 compareOffset = 3 compareDent = 4 19			Deput is 00h	
compareOffset = 3		•	Result is out	
19		compareOffset = 3		
compareBuffer = 5				
S5 55 55 22 33 45 F5 55 55 55 55	19			
Compare buffers with same parameters				
CompareBuffer = 55 55 52 2 33 43 F5 55 55 55 55 Compare buffers with same parameters Initialise compareBuffer CompareBuffer = 83 81 55 55 55 55 55 55 55 55 55 55 55 55 55			Result is -1	
S5 55 52 23 34 34 F5 55 55 55 55 Compare buffers with same parameters Initialise compareBuffer 83 81 55 55 55 55 55 55 55 55 55 FindAndCompareValue()	20			
Compare buffers with same parameters				
Initialise compareBuffer State S			Result is ±1	
CompareBuffer	21		INESUR IS TI	
findAndCompareValue() tag = 02h, occurrence = 1 valueOffset = 0 compareDeffset = 0 compareDeffset = 0 compareDeffset = 2 Initialise compareBuffer compareDeffer = 22 44 55 55 55 55 55 55 55 55 55 55 55 55		•		
tag = 02h, occurrence = 1 valueOffset = 0 compareOffset = 0 compareOffset = 0 compareLength = 2 22				
valueOffset = 0		findAndCompareValue()	Result is 00h	
compareOffset = 0				
Initialise compareBuffer CompareBuffer CompareBuffer 22 44 55 55 55 55 55 55 55 55 55 55 55 55				
compareBuffer = 22 44 55 55 55 55 55 55 55 55 55 55 55 55				
Result is 00h Result is 00h	22			
findAndCompareValue() tag = 02h, occurrence = 2 valueOffset = 0 compareOffset = 0 compareLength = 2 23 Initialise compareBuffer compareBuffer = 22 45 55 55 55 55 55 55 55 55 findAndCompareValue() tag = 02h, occurrence = 2 valueOffset = 0 compareOffset = 0 compareLength = 2 24 Initialise compareBuffer compareLength = 2 24 Initialise compareBuffer compareBuffer = 83 81 55 55 55 55 55 55 55 55 Successful call (with tag 02h) tag = 02h, occurrence = 1				
<pre>valueOffset = 0 compareOffset = 0 compareLength = 2 23</pre>			Result is 00h	
compareOffset = 0		9 ,		
compareLength = 2				
compareBuffer = 22 45 55 55 55 55 55 55 55 55 55 55 55 55		-		
22 45 55 55 55 55 55 55 55 55 55 55 findAndCompareValue() tag = 02h, occurrence = 2 valueOffset = 0 compareOffset = 0 compareLength = 2 24 Initialise compareBuffer compareBuffer = 83 81 55 55 55 55 55 55 55 55 Successful call (with tag 02h) tag = 02h, occurrence = 1	23			
findAndCompareValue() tag = 02h, occurrence = 2 valueOffset = 0 compareOffset = 0 compareLength = 2 24 Initialise compareBuffer compareBuffer = 83 81 55 55 55 55 55 55 55 55 Successful call (with tag 02h) tag = 02h, occurrence = 1				
tag = 02h, occurrence = 2 valueOffset = 0 compareOffset = 0 compareLength = 2 24			Pacult is -1	
compareOffset = 0			Nesult is - i	
compareLength = 2 24				
24				
compareBuffer =	24			
Successful call (with tag 02h) tag = 02h, occurrence = 1				
tag = 02h, occurrence = 1				
			Result is 00h	
		tag = U2n, occurrence = 1 valueOffset = 0		
compareBuffer.length = 12		compareBuffer.length = 12		
compareOffset = 0 compareLength = 2				
25 Initialise compareBuffer	25			
compareBuffer = 01 02 C4				
Successful call (with tag B3h) Result is 00h		Successful call (with tag B3h)	Result is 00h	
tag = B3h, occurrence = 1 valueOffset = 0				
valueOffset = 0				
compareOffset = 0		compareOffset = 0		
compareLength = 00C4	- 00		Deput of find And Course with A	
26 Successful call, findAndCompareValue with length =0 Result of findAndCompareValue() is 00h	26	•	•	
DstBuffer.length = C4			13 0011	
DstOffset = C4		DstOffset = C4		
DstLength = 0		•		
Send Formatted SMS PP with the maximum user data length = 0x010D, using 2				
concatenated envelopes				

ld	Description	API Expectation	APDU Expectation
	Initialise compareBuffer		
	CompareBuffer = 23.048 Header + secured		
	data (01 02 FA)		
27	Successful call (with SMS TPDU TAG)	Result is 00h	
	tag = 0Bh, occurrence = 1		
	valueOffset = 0x11		
	compareBuffer.length = 0x010D		
	compareOffset = 0		
	compareLength = 0x010D		
	Initialise compareBuffer		
	CompareBuffer = 55 55 55 EE EF F0 F1		
	F2 F3 F4 F5 F6 F7 F8 F9 FA		
28	Successful call (with SMS TPDU TAG)	Result is 00h	
	tag = 0Bh, occurrence = 1		
	valueOffset = 0x11		
	compareBuffer.length = 0x010D		
	compareOffset = 0x0100		
	compareLength = 0x0D		

6.2.4.17.4 Test Coverage

CRR number	Test case number
N1	12, 13
N2	15
N3	14, 18, 21, 22, 26, 27, 28
N4	16, 20
N5	17, 19, 23
N6	24, 25
P1	1
P2	2, 3, 4, 5, 6
P3	7, 8, 9, 10
P4	11
C1	Does not apply for EnvelopeHandler

6.2.4.18 Method getCapacity

Test Area Reference: API_2_ENH_GCAP

6.2.4.18.1 Conformance Requirement:

The method with following header shall be compliant to its definition in the API.

public byte getCapacity()

6.2.4.18.1.1 Normal execution

• CRRN1: The method shall return the maximum size of the Simple TLV list managed by the handler.

6.2.4.18.2 Test suite files

Test Script: API_2_ENH_GCAP_1.scr

Test Applet: API_2_ENH_GCAP_1.java

Load Script: API_2_ENH_GCAP_1.ldr

Cleanup Script: API_2_ENH_GCAP_1.clr

Parameter File: API_2_ENH_GCAP_1.par

6.2.4.18.3 Test Procedure

ld	Description	API Expectation	APDU Expectation
1	EnvelopeHandler available		
	Send envelope SMS-PP Formatted The applet calls the getLength() method The applet calls getCapacity()method	1 - Applet is triggered 2 - No exception is thrown 3 - No exception is thrown; the capacity is greater than the BER TLV Length	

6.2.4.18.4 Test Coverage

CRR number	Test case number
N1	1

6.2.4.19 Method getUserDataLength

Test Area Reference: API_2_ENH_GUDL

6.2.4.19.1 Conformance Requirement:

The method with following header shall be compliant to its definition in the API.

public short getUserDataLength()

6.2.4.19.1.1 Normal execution

- CRRN1: The method shall return the length of the User Data contained in the SMS TPDU TLV element.
- CRRN2: The length is from the first SMS TPDU TLV element.
- CRRN3: If the SMS TPDU TLV element is available, it becomes the selected TLV
- CRRN4: The method can be used if the event is EVENT_FORMATTED_SMS_PP_ENV.
- CRRN5: The method can be used if the event is EVENT FORMATTED SMS PP UPD.
- CRRN6: The method can be used if the event is EVENT_UNFORMATED_SMS_PP_ENV.
- CRRN7: The method can be used if the event is EVENT_UNFORMATTED_SMS_PP_UDP.

6.2.4.19.1.2 Context errors

- CRRC1: The method shall throw UNAVAILABLE_ELEMENT in case of unavailable TPDU TLV element.
- CRRC2: The method shall throw UNAVAILABLE_ELEMENT in case of wrong data format.

6.2.4.19.2 Test suite files

Specific triggering:

- UNFORMATTED_SMS_PP_ENV
- FORMATTED_SMS_PP_UPD
- UNFORMATED_SMS_PP_UPD
- UNRECOGNIZED_ENVELOPE
- For Formatted triggering if CC/RC/DS is used, the security parameters are those used for downloading applications.

Test Script: API_2_ENH_GUDL_1.scr

Test Applet: API_2_ENH_GUDL_1.java

Load Script: API_2_ENH_GUDL_1.ldr

Cleanup Script: API_2_ENH_GUDL_1.clr

Parameter File: API_2_ENH_GUDL_1.par

6.2.4.19.3 Test Procedure

ld	Description	API Expectation	APDU Expectation
	FORMATTED SMS PP ENV Triggering	·	-
1	Test with FORMATTED_SMS_PP_ENV and TP-	Returns 0x003D	
	OA length of 2 and user data length of 0x3D	D - t 0 - 000 D	
2	Test with TP-OA length of 12 and user data length of 0x3D	Returns 0x003D	
3	Test with RC/CC/DS length of 0 and secured data	Returns 0x0023	
	length of 0x10		
4		Returns 0x002B	
	length of 0x10		
5	Test with PCNTR = 0, no RC/CC/DS and data	Returns 0x0023	
	length of 0x10		
6	Test with PCNTR = 7, no RC/CC/DS and data	Returns 0x001F	
	length of 0x05		
7	Test with SecuredDataLength = 00 and no	Returns 0x0013	
	RC/CC/DS		
8	Test with UserDataLength = 0x7F	Returns 0x007F	
9	Test with UserDataLength = 0x80	Returns 0x0080	
10	Test with UserDataLength = maximum length	Returns 0x008C	
	(0x8C) for a single SMS		
11	Verify it is the first TPDU TLV:	Returns 0x0018	
	Send a SMS PP with 2 TPDU TLV with two		
	different user data lengths: 0x18 and 0x23		
12	Send envelope SMS-PP Formatted.	GetValueByte() returns 0x40(23.040	
	FindTLV() with TAG_DEVICE_IDENTITIES.	first byte)	
	GetUserDataLength() and then getValueByte()		
	with offset 0		
13	Test with UserDataLength = 0xFF with 2	Returns 0x00FF	
	concatenated SMS		
14	Test with UserDataLength = 0x100 with 2	Returns 0x0100	
	concatenated SMS		
15	Test with UserDataLength = maximum length	Returns 0x010D	
	(0x010D) with 2 concatenated SMS		
	FORMATTED SMS PP UPD Triggering		
16	Test with FORMATTED_SMS_PP_UPD and TP-	Returns 0x003D	
	OA length of 2 and user data length of 0x3D		
17		Returns 0x003D	
40	of 0x3D	D	
18		Returns 0x0023	
40	length of 0x10	D - t 0 - 000D	
19		Returns 0x002B	
20	length of 0x10	Poturno 0v0022	
20	Test with PCNTR = 0, no RC/CC/DS and data length of 0x10	Returns 0x0023	
21	Test with PCNTR = 7, no RC/CC/DS and data	Returns 0x001F	
21	length of 0x05	INGIGITIS UNUUTI	
22	Test with SecuredDataLength = 00 and no	Returns 0x0013	
~~	RC/CC/DS	TOTALING UNDO 10	
23	Test with UserDataLength = 0x7F	Returns 0x007F	
24	Test with UserDataLength = 0x80	Returns 0x0080	
	Test with UserDataLength = maximum	Returns 0x008C	
23	length(0x8C) for a single SMS	Totaliio oxooo	
26	Verify it is the first TPDU TLV:	Returns 0x0018	
	Send a SMS PP with 2 TPDU TLV with two	1333110 0700 10	
	different user data lengths: 0x18 and 0x23		
	, <u> </u>		

		, , , , , , , , , , , , , , , , , , , ,	
27	Send envelope SMS-PP Formatted.	GetValueByte() returns 0x40(23.040	
	FindTLV() with TAG_DEVICE_IDENTITIES.	first byte)	
	GetUserDataLength() and then getValueByte()		
	with offset 0		
28	Test with UserDataLength = 0xFF with 2	Returns 0x00FF	
	concatenated SMS		
29	Test with UserDataLength = 0x100 with 2	Returns 0x0100	
23	concatenated SMS	returns oxoroo	
30	Test with UserDataLength = maximum length	Returns 0x010D	
30		Retuins 0x010D	
	(0x010D) with 2 concatenated SMS		
	UNFORMATTED SMS PP ENV Triggering	D	
31	Test with UNFORMATTED_SMS_PP_ENV and	Returns 0x003D	
	TP-OA length of 2, and user data length of 0x3D		
32	Test with TP-OA length of 12, and user data length	Returns 0x003D	
	of 0x3D		
	Test with UserDataLength = 0x00	Returns 0x0000	
34	Test with UserDataLength = 0x7F	Returns 0x007F	
35	Test with UserDataLength = 0x80	Returns 0x0080	
36	Test with UserDataLength = maximum length:	Returns 0x008C	
	0x8C for a single SMS		
37	Verify it is the first TPDU TLV:	Returns 0x0018	
0,	Send a SMS PP with 2 TPDU TLV with two	returns oxoo ro	
	different user data lengths: 0x18 and 0x23		
38	Send envelope SMS-PP Unformatted.	GetValueByte() returns 0x00	
30	FindTLV() with TAG_DEVICE_IDENTITIES.	(23.040 first byte)	
	GetUserDataLength() and then getValueByte()	(23.040 first byte)	
	with offset 0 (first user data = 0x55)		
	UNFORMATTED SMS PP UPD Triggering		
39	Test with UNFORMATTED_SMS_PP_UPD and	Returns 0x003D	
	TP-OA length of 2, and user data length of 0x3D		
40	Test with TP-OA length of 12, and user data length	Returns 0x003D	
	of 0x3D		
41	Test with UserDataLength = 0x00	Returns 0x0000	
42	Test with UserDataLength = 0x7F	Returns 0x007F	
43	Test with UserDataLength = 0x80	Returns 0x0080	
44	Test with UserDataLength = maximum length:	Returns 0x008C	
	0x8C for a single SMS		
45	Verify it is the first TPDU TLV:	Returns 0x0018	
.	Send a SMS PP with 2 TPDU TLV with two		
	different user data lengths: 0x18 and 0x23		
46	Send envelope SMS-PP Unformatted.	GetValueByte() returns 0x00	
40	FindTLV() with TAG_DEVICE_IDENTITIES.	(23.040 first byte)	
		(23.040 IIISt Dyte)	
	GetUserDataLength() and then getValueByte()		
	with offset 0		
L	UNRECOGNIZED_ENVELOPE Triggering		
47	Test with an UNRECOGNIZED_ENVELOPE	ToolkitException	
1		UNAVAILABLE_ELEMENT	

6.2.4.19.4 Test Coverage

CRR number	Test case number
N1	All test cases excepted:
	53
N2	11, 26, 37, 45
N3	12, 27, 38, 46
N4	1 to 15
N5	16 to 30
N6	31 to 38
N7	39 to 46
C1	47
C2	Not applicable

6.2.4.20 Method getChannelldentifier

Test Area Reference: API_2_ENH_GCID

6.2.4.20.1 Conformance Requirement:

The method with following header shall be compliant to its definition in the API.

6.2.4.20.1.1 Normal execution

- CRRN1: The method shall return the channel identifier byte value.
- CRRN2: The channel identifier byte value returned shall be from the first Channel status TLV element.
- CRRN3: If the element is available it becomes the currently selected TLV.
- CRRN4: The channel identifier is available for all triggered toolkit applets from the invocation to the termination of their processToolkit method if the EnvelopeHandler is available.

6.2.4.20.1.2 Context errors

- CRRC1: The method shall throw ToolkitException (UNAVAILABLE_ELEMENT) if the Channel status TLV is not present.
- CRRC2: The method shall throw ToolkitException (OUT_OF_TLV_BOUNDARIES) if the Simple TLV Channel Status length is equal to 0.

6.2.4.20.2 Test suite files

Test Script: API_2_ENH_GCID_1.scr

Test Applet: API_2_ENH_GCID_1.java

Load Script: API_2_ENH_GCID_1.ldr

Cleanup Script: API_2_ENH_GCID_1.clr

Parameter File: API_2_ENH_GCID_1.par

6.2.4.20.3 Test Procedure

ld	Description	API Expectation	APDU Expectation
0	1- Applet1 is installed with maximum number of channel = 07. 2- Applet1 builds proactive commands OPEN CHANNEL with init() method in order to open all channels. ProactiveHandler.send() methods are called.		2- OPEN CHANNEL proactive command is fetched TERMINAL RESPONSE is issued with Channel Id from 01 to 07
1	1- Send envelope Event Download Channel Status with channel status TLV: channel status value = 0x8100. 2- Call EnvelopeHandler.getChannelIdentifier() method	1- Applet1 is triggered 2- Returns 0x01	
2	1- Send envelope Event Download Channel Status with two channel status TLV: first value = 0x8400 second value = 0x8500. 2- Call twice the EnvelopeHandler.getChannelIdentifier() method	2- Returns 0x04 Returns 0x04	
3	1- Send envelope Event Download Channel Status with channel status TLV: Channel Status value = 0x0605 ViewHandler.FindTLV() with Device IdentityTag. 2- Call EnvelopeHandler.getChannelIdentifier() method. 3- Compare EnvelopeHandler.getChannelIdentifier() and then ViewHandler.getValueByte(0).	2- Returns 0x06 3- GetChannelldentifier() =getValueByte(0)	
4	 Send envelope Menu Selection without Channel Status TLV. Call EnvelopeHandler.getChannelIdentifier() method. 	2- A Toolkit exception UNAVAILABLE_ELEMEN T is thrown.	
5	1- Send Envelope Event Download Channel Status with Channel Status TLV: Channel status value = 0x0600 2- Call EnvelopeHandler.getChannelIdentifier() method.	1- Returns 0x06	
6	1- Send unrecognized envelope with a Channel Status TLV having a length equal to 0. 2- Call EnvelopeHandler.getChannelIdentifier() method.	2- A Toolkit exception OUT_OF_TLV_BOUNDA RIES is thrown.	

6.2.4.20.4 Test Coverage

CRR number	Test case number
N1	1, 2
N2	3
N3	3
N4	5
C1	4
C2	6

6.2.5 Class EnvelopeResponseHandler

6.2.5.1 Method getTheHandler

Test Area Reference: API_2_ERH_GTHD

6.2.5.1.1 Conformance requirement:

The method with following header shall be compliant to its definition in the API.

6.2.5.1.1.1 Normal execution

- CRRN1: The method shall return the single system instance of the EnvelopeResponseHandler class.
- CRRN2: The EnvelopeResponseHandler is a Temporary JCRE Entry Point Object (see Javacard 2.1 Runtime Environment (JCRE) Specification [12]).

6.2.5.1.1.1 Parameter errors

No requirements.

6.2.5.1.1.3 Context errors

- CRRC1: The method shall thrown ToolkitException (HANDLER_NOT_AVAILABLE) if the handler is busy.
- CRRC2: After the first invocation of the ProactiveHandler.send method the EnvelopeResponseHandler is no more available

6.2.5.1.2 Test suite files

Test Script: API_2_ERH_GTHD_1.scr

Test Applet: API_2_ERH_GTHD_1.java

Load Script: API_2_ERH_GTHD_1.ldr

Cleanup Script: API_2_ERH_GTHD_1.clr

Parameter File: API_2_ERH_GTHD_1.par

6.2.5.1.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	getTheHandler twice	The returned objects shall be the	
		same	
2	Verify that getTheHandler returns an	The reference returned shall be an	
	EnvelopeHandler	EnvelopeResponseHandler	
	getTheHandler	(checkcast)	
3	Verify the returned value is not null	The reference returned shall not be	
	getTheHandler	null.	
4	getTheHandler, then send a proactive	ToolkitException	
	command, and then, appendTLV	HANDLER_NOT_AVAILABLE is	
		thrown	

6.2.5.1.4 Test Coverage

CRR number	Test case number
N1	1, 2, 3
N2	To be checked in Framework tests and insert here cross reference
C1	To be checked in Framework tests and insert here cross reference
C2	4

6.2.5.2 Method post

Test Area Reference: API_2_ERH_POSTB

6.2.5.2.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

6.2.5.2.1.1 Normal execution

- CRRN1: When the method is called, the toolkit applet can continue it's processing (e.g. prepare a proactive command).
- CRRN2: The byte statusType is SW1 of the status.
- CRRN3: If the send method is called after a post method, the posted data are the first sent to the ME.
- CRRN4: The SIM Toolkit Framework shall take the optional Application Data posted by the triggered toolkit applet if present, secure and send the response packet the SIM Toolkit Framework will return the response APDU defined by the toolkit applet.

6.2.5.2.1.2 Parameter error

No requirements.

6.2.5.2.1.3 Context errors

• CRRC1: The method shall thrown ToolkitException (HANDLER_NOT_AVAILABLE) if the handler is busy.

6.2.5.2.2 Test suite files

Specific triggering: Call control

Test Script: API_2_ERH_POSTB_1.scr

Test Applet: API_2_ERH_POSTB_1.java

Load Script: API_2_ERH_POSTB_1.ldr

Cleanup Script: API_2_ERH_POSTB_1.clr

Parameter File: API_2_ERH_POSTB_1.par

6.2.5.2.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	getTheHandler and then post		9000
	(the handler is empty)		
2	Fill the handler (appendTLV to have bytes		9FFD data are retrieved with GET
	in it)and then post data with status		RESPONSE command
	9F		
3	Verify that after a post the handler is no	ToolkitException	
	more available	HANDLER_NOT_AVAILABL	
	appendTLV, post and then appendTLV	E is thrown on the second	
		appendTLV	
4	construct the response (appendTLV with		9E12 and posted data retrieved by a
	0x10 data) and post it with status 9E and		GET RESPONSE with status 9113
	then send a display text		and display text retrieved by a FETCH
5	Verify that it is possible to send a proactive		91 13 and display text is retrieved by
	command after a post		a FETCH
	getTheHandler and post , then send a		
	display text		
6	Verify it is not possible to post after a	ToolkitException	
	proactive command	HANDLER_NOT_AVAILABL	
	getTheHandler, appendTLV, send a	E is thrown	
	display text, post.		
7	Verify that the handler is no more available		9E12 and posted data retrieved by a
	after a post	HANDLER_NOT_AVAILABL	GET RESPONSE
	getTheHandler, appendTLV, post with	E is thrown	

S	tatus 9E, post with status 9F	

6.2.5.2.4 Test Coverage

CRR number	Test case number
N1	3, 4, 7
N2	1, 2, 4, 7
N3	4, 5
N4	To be checked in Framework tests and insert here cross reference
C1	6

6.2.5.3 Method postAsBERTLV

Test Area Reference: API_2_ERH_POSTBB

6.2.5.3.1 Conformance Requirement

The method with following header shall be compliant to its definition in the API.

6.2.5.3.1.1 Normal execution

- CRRN1: When the method is called, the toolkit applet can continue it's processing (e.g. prepare a proactive command) the SIM Toolkit Framework will return the response APDU defined by the toolkit applet.
- CRRN2: The byte statusType is SW1 of the status
- CRRN3: If the send method is called after a postAsBERTLV method, the posted data are the first sent to the MF
- CRRN4: The byte tag is the BER Tag at the beginning of the simple TLV list.

6.2.5.3.1.2 Parameter errors

No requirements.

6.2.5.3.1.3 Context errors

• CRRC1: The method shall thrown ToolkitException (HANDLER_NOT_AVAILABLE) if the handler is busy.

6.2.5.3.2 Test suite files

Specific triggering: Call control

Test Script: API_2_ERH_POSTBB_1.scr

Test Applet: API_2_ERH_POSTBB_1.java

Load Script: API_2_ERH_POSTBB_1.ldr

Cleanup Script: API_2_ERH_POSTBB_1.clr

Parameter File: API_2_ERH_POSTBB_1.par

6.2.5.3.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	getTheHandler and then postAsBERTLV		9F02 data are retrieved with
	(the handler is empty)		GET RESPONSE command,

ld	Description	API Expectation	APDU Expectation
			the tag shall be 33 and the length is 00
2	Fill the handler and then postAsBERTLV the data with status 9F, and tag 33		9FFF data are retrieved with GET RESPONSE command, the tag shall be 33
3	appendTLV, postAsBERTLV and then appendTLV	ToolkitException HANDLER_NOT_AVAILABLE is thrown on the second appendTLV	
4	construct the response (appendTLV with 0x10 data) and postAsBERTLV it with status 9E, tag 75 and then send a display text		9E14 and posted data retrieved by a GET RESPONSE the tag shall be 75 with status 9113 and display text retrieved by a FETCH
5	getTheHandler and postAsBERTLV, then send a display text		9E02 and posted data retrieved by a GET RESPONSE the tag 33 (and the length 00) with status 9113 and display text is retrieved by a FETCH
6	Verify it is not possible to postAsBERTLV after a proactive command getTheHandler, appendTLV, send a display text, postAsBERTLV.	ToolkitException HANDLER_NOT_AVAILABLE is thrown on the postAsBERTLV	
7	Verify that the handler is no more available after a postAsBERTLV getTheHandler, appendTLV(with data length = 0x10, postAsBERTLV with status 9E, tag 56, postAsBERTLV with status 9F, tag 28	ToolkitException HANDLER_NOT_AVAILABLE is thrown on the second postAsBERTLV	9E14 and posted data retrieved by a GET RESPONSE the tag shall be 56 with status 9000

6.2.5.3.4 Test Coverage

CRR number	Test case number
N1	3, 4, 7
N2	1, 2, 4, 7
N3	4, 5
N4	2, 4, 7
C1	6

6.2.5.4 Method getLength

Test Area Reference: API_2_ERH_GLEN

6.2.5.4.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

6.2.5.4.1.1 Normal execution

• CRRN1: returns the length in bytes of the TLV list.

6.2.5.4.1.2 Parameter errors

No requirements.

6.2.5.4.1.3 Context errors

• CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException.HANDLER_NOT_AVAILABLE.

6.2.5.4.2 Test Suite files

Specific triggering: Unrecognized Envelope

Test Script: API_2_ERH_GLEN_1.scr

Test Applet: API_2_ERH_GLEN_1.java

Load Script: API_2_ERH_GLEN_1.ldr

Cleanup Script: API_2_ERH_GLEN_1.clr

Parameter File: API_2_ERH_GLEN_1.par

6.2.5.4.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Clear the handler	Result of getLength() is 0	
	getLength()		
2	appendTLV with length of 7	Result of getLength() is 9	
	getLength()		
3	Clear the handler and appendTLV with Length	Result of getLength() is 253	
	of 250		
	getLength()		
4	Build a 7Fh Envelope response handler	Result of getLength() is 81h	
	getLength()		
5	Build a 80h Envelope response handler	Result of getLength() is 83h	
	getLength()		
NOT	NOTE: Test case 3 is limited to 253 and not 256 because the current 3GPP TS 43.019 [7] is not clear enough on this		

NOTE: Test case 3 is limited to 253 and not 256 because the current 3GPP TS 43.019 [7] is not clear enough on this point. So this test allows the two possible implementations.

6.2.5.4.4 Test Coverage

CRR number	Test case number
N1	1, 2, 3, 4, 5
C1	Does not apply for Envelope response handler

6.2.5.5 Method copy

Test Area Reference: API_2_ERH_COPY_BSS

6.2.5.5.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

6.2.5.5.1.1 Normal execution

- CRRN1: copies the simple TLV list contained in the handler to the destination byte array.
- CRRN2: returns dstOffset + dstLength.

6.2.5.5.1.2 Parameter errors

- CRRP1: if dstBuffer is null a NullPointerException is thrown.
- CRRP2: if dstOffset or dstLength or both would cause access outside array bounds, or if dstLength is negative, an ArrayIndexOutOfBoundsException is thrown.
- CRRP3: if dstLength is greater than the length of the simple TLV List, an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException. OUT_OF_TLV_BOUNDARIES.

6.2.5.5.1.3 Context errors

• CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.

6.2.5.5.2 Test Suite files

Specific triggering: Unrecognized Envelope

Test Script: API_2_ERH_COPY_BSS_1.scr

Test Applet: API_2_ERH_COPY_BSS_1.java

Load Script: API_2_ERH_COPY_BSS_1.ldr

Cleanup Script: API_2_ERH_COPY_BSS_1.clr

Parameter File: API_2_ERH_COPY_BSS_1.par

6.2.5.5.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	appendTLV with value length of 7		
	NULL as parameter to dstBuffer	NullPointerException is thrown	
2			
	dstOffset ≥ dstBuffer.length	ArrayIndexOutOfBoundsExceptio	
	dstBuffer.length = 5	n is thrown	
	dstOffset = 5		
3	dstLength = 1 dstOffset < 0	ArrayladayOutOfPauadaEyaantia	
3	dstBuffer.length = 5	ArrayIndexOutOfBoundsException is thrown	
	dstOffset = -1	II IS UIIOWII	
	dstLength = 1		
4	dstLength > dstBuffer.length	ArrayIndexOutOfBoundsExceptio	
	dstBuffer.length = 5	n is thrown	
	dstOffset = 0		
_	dstLength = 6	A d- d O- +O+D d- F ti-	
5	dstOffset + dstLength > dstBuffer.length dstBuffer.length = 5	ArrayIndexOutOfBoundsExceptio	
	dstOffset = 3	n is thrown	
	dstLength = 3		
6	dstLength < 0	ArrayIndexOutOfBoundsExceptio	
	dstBuffer.length = 5	n is thrown	
	dstOffset = 0		
	dstLength = -1	T HE CONTROL THE	
7	dstLength > length of the simple TLV list	ToolkitException.OUT_OF_TLV_	
	<pre>dstBuffer.length = 10 dstOffset = 0</pre>	BOUNDARIES is thrown	
	dstLength = 10		
8	Successful call, dstBuffer is the whole buffer	Result of copy() is 9	
	dstBuffer.length = 9	1170	
	dstOffset = 0		
	dstLength = 9		
9	Compare the buffer	Result of arrayCompare() is 0	
10	Successful call, dstBuffer is part of a buffer	Result of copy() is 12	
	<pre>dstBuffer.length = 15 dstOffset = 3</pre>		
	dstLength = 9		
11	Compare the whole buffer	Result of arrayCompare() is 0	
12	Successful call, dstBuffer is part of a buffer	Result of copy() is 9	
'-	dstBuffer.length = 15	1 (3 (3) (3) (3) (3) (3) (3) (3) (3) (3)	
•	1	ı ,	

	dstOffset = 3	
	dstLength = 6	
13	Compare the whole buffer	Result of arrayCompare() is 0
14	Successful call, copy with length =0	Result of copy() is 15
	dstBuffer.length = 15	
	dstOffset = 15	
	dstLength = 0	

6.2.5.5.4 Test Coverage

CRR number	Test case number
N1	9, 11, 13
N2	8, 10, 12, 14
P1	1
P2	2, 3, 4, 5, 6
P3	7
C1	Does not apply for Envelope response handler

6.2.5.6 Method findTLV

Test Area Reference: API_2_ERH_FINDBB

6.2.5.6.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

6.2.5.6.1.1 Normal execution

Looks for the indicated occurrence of a TLV element from the beginning of the TLV list (handler buffer):

- CRRN1: the method is successful if the required occurrence exists then the corresponding TLV becomes current.
- CRRN2: if the method is successful then it returns TLV_FOUND_CR_SET when Comprehension Required flag is set.
- CRRN3: if the method is successful then it returns TLV_FOUND_CR_NOT_SET when Comprehension Required flag is not set.
- CRRN4: if the required occurrence of the TLV element does not exist, the current TLV is no longer defined and TLV_NOT_FOUND is returned.
- CRRN5: The search method is comprehension required flag independent.

6.2.5.6.1.2 Parameter errors

• CRRP1: if an input parameter is not valid (e.g. occurrence = 0) an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException BAD_INPUT_PARAMETER.

6.2.5.6.1.3 Context errors

• CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.

6.2.5.6.2 Test Suite files

Specific triggering: Unrecognized Envelope

Test Script: API_2_ERH_FINDBB_1.scr

Test Applet: API_2_ERH_FINDBB_1.java

Load Script: API_2_ERH_FINDBB_1.ldr

Cleanup Script: API_2_ERH_FINDBB_1.clr

Parameter File: API_2_ERH_FINDBB_1.par

6.2.5.6.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	append the handler with TLVs:		
	81 03 11 22 33		
	82 02 99 77	ToolkitEveention DAD INDUT DA	
	Invalid input parameter Occurrence = 0	ToolkitException.BAD_INPUT_PA	
	occurrence = 0	RAIVIETER IS THROWN	
2			
	Search 1st TLV	Result is TLV_FOUND_CR_SET	
	Tag = 01h		
	Occurrence = 1		
3	Call the getValueLength() method	Result is 03h	
4	Search 2nd TLV	Result is TLV_FOUND_CR_SET	
	Tag = 02h Occurrence = 1		
5	Call the getValueLength() method	Result is 02h	
6	Select a TLV (tag 02h)	ixesuit is 02ii	
	Search a wrong tag	Result is TLV_NOT_FOUND	
	Tag = 03h	Tresult is TEV_IVOT_I COIVE	
	Occurrence = 1		
7	Call the getValueLength() method	ToolkitException.UNAVAILABLE_	
		ELEMENT is thrown.	
8	Search a tag with wrong occurrence	Result is TLV_NOT_FOUND	
	Tag = 01h		
_	Occurrence = 2	TU-4F	
9	Call the getValueLength() method	ToolkitException.UNAVAILABLE_ ELEMENT is thrown.	
10	Append a TLV with tag=02h	ELEMENT IS UTOWIT.	
10	Search the TLV	Result is	
	Tag = 02h	TLV_FOUND_CR_NOT_SET	
	Occurrence = 2	TEV_1 OUND_ON_NOT_OE1	
11	Append a TLV with tag=04h		
	Search the TLV	Result is	
	Tag = 04h	TLV_FOUND_CR_NOT_SET	
10	Occurrence = 1	D 1: TIV FOLIND OF 3	
12	Search tag 81h	Result is TLV_FOUND_CR_SET	
	Tag = 81h Occurrence = 1		
13	Search tag 84h	Result is	
	Tag = 84h	TLV_FOUND_CR_NOT_SET	
	Occurrence = 1	.2	

6.2.5.6.4 Test Coverage

CRR number	Test case number
N1	3, 5
N2	2, 4
N3	10, 11
N4	6, 7,8, 9
N5	12, 13
P1	1
C1	Does not apply for Envelope response handler

6.2.5.7 Method getValueLength

Test Area Reference: API_2_ERH_GVLE

6.2.5.7.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

6.2.5.7.1.1 Normal execution

• CRRN1: gets and returns the binary length of the value field for the last TLV element which has been found in the handler.

6.2.5.7.1.2 Parameter errors

No requirements.

6.2.5.7.1.3 Context errors

- CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.
- CRRC2: in case of unavailable TLV element an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException UNAVAILABLE_ELEMENT.

6.2.5.7.2 Test Suite files

Specific triggering: Unrecognized Envelope

Test Script: API_2_ERH_GVLE_1.scr

Test Applet: API_2_ERH_GVLE_1.java

Load Script: API_2_ERH_GVLE_1.ldr

Cleanup Script: API_2_ERH_GVLE_1.clr

Parameter File: API_2_ERH_GVLE_1.par

6.2.5.7.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	appendTLV 02 02 02 02	-	<u>-</u>
	findTLV with TAG 03		
	getValueLength()	ToolkitException.UNAVAILABLE_ ELEMENT is thrown	
2	appendTLV with TAG 0D and length 00		
	Search TLV 0Dh		
	getValueLength()	Result is 00h	
3	Clear the handler and append TLV with TAG 0D and length 02		
	Search TLV 0Dh		
	getValueLength()	Result is 02h	
4	Clear the handler and append TLV with TAG 0D and length 0x7F		
	Search TLV 0Dh		
	getValueLength()	Result is 7Fh	
5	Clear the handler and append TLV with TAG 0D and length 0x80		
	Search TLV 0Dh		
	getValueLength()	Result is 80h	
6	Clear the handler and append TLV with TAG		

0D and length 0xF1		
Search TLV 0Dh		
<pre>getValueLength()</pre>	Result is F1h	

6.2.5.7.4 Test Coverage

CRR number	Test case number
N1	2, 3, 4, 5, 6
C1	Does not apply for EnvelopeResponseHandler
C2	1

6.2.5.8 Method getValueByte

Test Area Reference: API_2_ERH_GVBYS

6.2.5.8.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

6.2.5.8.1.1 Normal execution

• CRRN1: Gets a byte from the last TLV element which has been found in the handler and returns its value (1 byte).

6.2.5.8.1.2 Parameter errors

• CRRP1: if valueOffset is out of the current TLV an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException OUT_OF_TLV_BOUNDARIES.

6.2.5.8.1.3 Context errors

- CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.
- CRRC2: in case of unavailable TLV element an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException UNAVAILABLE_ELEMENT.

6.2.5.8.2 Test Suite files

Specific triggering: Unrecognized Envelope

Test Script: API_2_ERH_GVBYS_1.scr

Test Applet: API_2_ERH_GVBYS_1.java

Load Script: API_2_ERH_GVBYS_1.ldr

Cleanup Script: API_2_ERH_GVBYS_1.clr

Parameter File: API_2_ERH_GVBYS_1.par

6.2.5.8.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	appendTLV 82 02 81 82, appendTLV 81 03 11		
	22 FE		
	findTLV with TAG 03		
	getValueByte(0)	ToolkitException.UNAVAILABLE_	

ld	Description	API Expectation	APDU Expectation
		ELEMENT is thrown	
2	Search TLV 01h		
	getValueByte(3)	ToolkitException.OUT_OF_TLV_ BOUNDARIES is thrown	
3	Search TLV 01h		
	getValueByte(2)	Result is FEh	
4	Search TLV 02h		
	getValueByte(0)	Result is 81h	
5	appendTLV with TAG 0D, Length 0x7E, Value: 00, 01,, 7D		
	getValueByte(7D)	Result is 7Dh	
6	clear the handler, appendTLV with TAG 0D, Length 0x80, Value: 00, 01,, 7F		
	getValueByte(7E)	Result is 7Eh	
7	getValueByte(7F)	Result is 7Fh	
8	clear the handler, appendTLV with TAG 0D, Length 0xF1, Value: 00, 01,, F0		
	getValueByte(F0)	Result is F0h	

6.2.5.8.4 Test Coverage

CRR number	Test case number
N1	3, 4, 5, 6, 7, 8
P1	2
C1	Does not apply for EnvelopeResponseHandler
C2	1

6.2.5.9 Method copyValue

Test Area Reference: API_2_ERH_CPYVS_BSS

6.2.5.9.1 Conformance requirement

The method with following header shall be compliant with its definition in the API.

6.2.5.9.1.1 Normal execution

- CRRN1: copies a part of the last TLV element which has been found, into a destination. buffer.
- CRRN2: returns dstOffset + dstLength.

6.2.5.9.1.2 Parameter errors

- CRRP1: if dstBuffer is null NullPointerException is thrown.
- CRRP2: if dstOffset or dstLength or both would cause access outside array bounds, or if dstLength is negative ArrayIndexOutOfBoundsException is thrown.
- CRRP3: if valueOffset, dstLength or both are out of the current TLV an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException OUT_OF_TLV_BOUNDARIES.

6.2.5.9.1.3 Context errors

- CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.
- CRRC2: in case of unavailable TLV element an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException UNAVAILABLE_ELEMENT.

6.2.5.9.2 Test Suite files

Specific triggering: Unrecognized Envelope

Test Script: API_2_ERH_CPYVS_BSS_1.scr

Test Applet: API_2_ERH_CPYVS_BSS_1.java

Load Script: API_2_ERH_CPYVS_BSS_1.ldr

Cleanup Script: API_2_ERH_CPYVS_BSS_1.clr

Parameter File: API_2_ERH_CPYVS_BSS_1.par

6.2.5.9.3 Test procedure

	Description	ADI Francotation	ADDII Francistica
ld	Description	API Expectation	APDU Expectation
1	appendTLV with TAG: 0D and length 16		
	Select Text String TLV		
	copyValue() with a null dstBuffer	NullPointerException is thrown	
2			
	dstOffset ≥ dstBuffer.length	ArrayIndexOutOfBoundsExceptio	
	dstBuffer.length = 5	n is thrown	
	dstOffset = 5		
	dstLength = 1		
3	dstOffset < 0	ArrayIndexOutOfBoundsExceptio	
	dstBuffer.length = 5	n is thrown	
	dstOffset = -1		
4	dstLength = 1	A movel and a viOvitOfD a via da Evica antica	
4	dstLength >dstBuffer.length dstBuffer.length = 5	ArrayIndexOutOfBoundsExceptio	
	dstOffset = 0	n is thrown	
	dstollset = 0 dstLength = 6		
5	dstOffset + dstLength >dstBuffer.length	ArrayIndexOutOfBoundsExceptio	
ັ	dstBuffer.length = 5	n is thrown	
	dstOffset = 3	II is unown	
	dstLength = 3		
6	dstLength < 0	ArrayIndexOutOfBoundsExceptio	
	dstBuffer.length = 5	n is thrown	
	dstOffset = 0		
	dstLength = -1		
7	clear the handler, appendTLV with TAG: 0D		
	and length 6		
	Select Text String TLV		
	valueOffset ≥ Text String Length	ToolkitException.OUT_OF_TLV_	
	valueOffset = 6	BOUNDARIES is thrown	
	dstBuffer.length = 15		
	dstOffset = 0		
<u> </u>	dstLength = 1		
8	valueOffset < 0	ToolkitException.OUT_OF_TLV_	
	valueOffset = -1	BOUNDARIES is thrown	
	dstBuffer.length = 15		
	dstOffset = 0		
9	dstLength = 1 dstLength > Text String length	ToolkitEveention OLIT OF TLV	
9	valueOffset = 0	ToolkitException.OUT_OF_TLV_	
	dstBuffer.length = 15	BOUNDARIES is thrown	
	dstOffset = 0		
	dstLength = 7		
10	valueOffset + dstLength > Text String length	ToolkitException.OUT_OF_TLV_	
'	valueOffset = 2	BOUNDARIES is thrown	
	dstBuffer.length = 15	DOG! ID/ II II DO IO II II OWI!	
	dstOffset = 0		
	astuliset = U		

113

ld	Description	API Expectation	APDU Expectation
	dstLength = 5		
11	Initialise the handler		
	copyValue()	ToolkitException.UNAVAILABLE_	
		ELEMENT is thrown	
12	clear the handler, appendTLV with TAG: 0D		
	and value: 04 00 01 0F		
	Select Text String TLV		
	Successful call	Result of copyValue() is 17	
	<pre>valueOffset = 0</pre>	.,	
	dstBuffer.length = 17		
	dstOffset = 0		
10	dstLength = 17	D. It is and	
13	Compare buffer	Result is 00h	
	buffer = 04 00 01 0F		
14	initialise dstBuffer		
14	dstBuffer = 55 55 55		
	Successful call	Result of copyValue() is 15	
	valueOffset = 2	itesuit of copyvalue() is 15	
	dstBuffer.length = 20		
	dstOffset = 3		
	dstLength = 12		
15	Compare buffer	Result is 00h	
	buffer =		
	55 55 55 01 02		
	03 04 05 06 07 08 09 0A 0B 0C		
	108 09 0A 0B 0C		
16	Successful call, copyValue with length =0	Result of copyValue() is 20	
10	dstBuffer.length = 20	result of copy value() is 20	
	dstOffset = 20		
1	dstLength = 0		

6.2.5.9.4 Test Coverage

CRR number	Test case number
N1	13, 15
N2	12, 14, 16
P1	1
P2	2, 3, 4, 5, 6
P3	7, 8, 9, 10
C1	Does not apply for EnvelopeResponseHandler
C2	11

6.2.5.10 Method compareValue

Test Area Reference: API_2_ERH_CPRVS_BSS

6.2.5.10.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

6.2.5.10.1.1 Normal execution

Compares the last found TLV element with a buffer:

• CRRN1: returns 0 if identical.

- CRRN2: returns -1 if the first miscomparing byte in simple TLV List is less than that in compareBuffer.
- CRRN3: returns 1 if the first miscomparing byte in simple TLV List is greater than that in compareBuffer.

6.2.5.10.1.2 Parameter errors

- CRRP1: if compareBuffer is null NullPointerException shall be thrown.
- CRRP2: if compareOffset or compareLength or both would cause access outside array bounds, or if compareLength is negative ArrayIndexOutOfBoundsException shall be thrown.
- CRRP3: if valueOffset, dstLength or both are out of the current TLV an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException OUT_OF_TLV_BOUNDARIES.

6.2.5.10.1.3 Context errors

- CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER NOT AVAILABLE.
- CRRC2: in case of unavailable TLV element an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException UNAVAILABLE_ELEMENT.

6.2.5.10.2 Test Suite files

Specific triggering: Unrecognized Envelope

Test Script: API_2_ERH_CPRVS_BSS_1.scr

Test Applet: API_2_ERH_CPRVS_BSS_1.java

Load Script: API_2_ERH_CPRVS_BSS_1.ldr

Cleanup Script: API_2_ERH_CPRVS_BSS_1.clr

Parameter File: API_2_ERH_CPRVS_BSS_1.par

6.2.5.10.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	appendTLV with TAG: 0D and length 16 Select Text String TLV		
	compareValue() with a null compareBuffer	NullPointerException is thrown	
2			
	<pre>compareOffset ≥ compareBuffer.length compareBuffer.length = 5 compareOffset = 5 compareLength = 1</pre>	ArrayIndexOutOfBoundsExceptio n is thrown	
3	<pre>compareOffset < 0 compareBuffer.length = 5 compareOffset = -1 compareLength = 1</pre>	ArrayIndexOutOfBoundsExceptio n is thrown	
4	<pre>compareLength >compareBuffer.length compareBuffer.length = 5 compareOffset = 0 compareLength = 6</pre>	ArrayIndexOutOfBoundsExceptio n is thrown	
5	<pre>compareOffset + compareLength</pre>	ArrayIndexOutOfBoundsException is thrown	
6	<pre>compareLength < 0 compareBuffer.length = 5 compareOffset = 0 compareLength = -1</pre>	ArrayIndexOutOfBoundsExceptio n is thrown	

ld	Description	API Expectation	APDU Expectation
7	appendTLV with TAG: 0D and length 6	7 1 = Apoctume	7 2 C Experience:
<i>'</i>	Select Text String TLV		
		ToolkitEveention OUT OF TIV	
	valueOffset ≥ Text String Length	ToolkitException.OUT_OF_TLV_	
	<pre>valueOffset = 6 compareBuffer.length = 15</pre>	BOUNDARIES is thrown	
	compareOffset = 0		
	compareLength = 1		
8	valueOffset < 0	ToolkitException.OUT_OF_TLV_	
0	valueOffset = -1	BOUNDARIES is thrown	
	compareBuffer.length = 15	DOONDAINEO IS UITOWIT	
	compareOffset = 0		
	compareLength = 1		
9	compareLength > Text String length	ToolkitException.OUT_OF_TLV_	
	valueOffset = 0	BOUNDARIES is thrown	
	compareBuffer.length = 15		
	<pre>compareOffset = 0 compareLength = 7</pre>		
10	valueOffset + compareLength > Text String	ToolkitException.OUT_OF_TLV_	
10	length	BOUNDARIES is thrown	
	valueOffset = 2	BOUNDARIES IS IIIIOWII	
	compareBuffer.length = 15		
	compareOffset = 0		
	compareLength = 5		
11	Initialise the handler		
	compareValue()	ToolkitException.UNAVAILABLE_	
		ELEMENT is thrown	
12	appendTLV with TAG: 0D and value: 04 00 01		
	0F		
	Select Text String TLV		
	Initialise compareBuffer		
	compareBuffer =		
	04 00 01 0F		
	Compare buffers	Result is 00h	
	<pre>valueOffset = 0 compareOffset = 0</pre>		
	compareLength = 17		
13	Initialise compareBuffer		
	compareBuffer =		
	04 00 01 02 03		
	04 05 06 07 08		
	05 0A 0B 0C 0D		
	0E 10	Decultie 4	
4.4	Compare buffers with same parameters	Result is -1	
14	Initialise compareBuffer compareBuffer =		
	03 00 01 OF		
	Compare buffers with same parameters	Result is +1	
15	Initialise compareBuffer		
	compareBuffer =		
	55 55 55 01 02		
	03 04 05 06 07		
	08 09 0A 0B 0C		
	55 55 55 55 55 Compare huffers	Result is 00h	
	Compare buffers valueOffset = 2	Leanii ia noii	
	compareOffset = 3		
	compareLength = 12		
16	Initialise compareBuffer		
	compareBuffer =		
	55 55 55 02 01		
	03 04 05 06 07 08 09 0A 0B 0C		
	55 55 55 55 55		
	Compare buffers with same parameters	Result is -1	
17	Initialise compareBuffer		
•	compareBuffer =		
	55 55 55 01 02		
	03 04 05 06 07		
	08 09 0A 0A 0D		
	55 55 55 55 55 Compare buffers with same parameters	Popult is 11	
	Compare buffers with same parameters	Result is +1	

ld	Description	API Expectation	APDU Expectation
18	Successful call, compareValue with length =0	Result of compareValue() is 0	
	compareBuffer.length = 15		
	compareOffset = 15		
	compareLength = 0		

6.2.5.10.4 Test Coverage

CRR number	Test case number
N1	12, 15, 18
N2	13, 16
N3	14, 17
P1	1
P2	2, 3, 4, 5, 6
P3	7, 8, 9, 10
C1	Does not apply for EnvelopeResponseHandler
C2	11

6.2.5.11 Method findAndCopyValue(byte tag, byte[] dstBuffer, short valueOffset)

Test Area Reference: API_2_ERH_FACYB_BS

6.2.5.11.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

6.2.5.11.1.1 Normal execution

- CRRN1: looks for the first occurrence of a TLV element from the beginning of a TLV list and copy its value into a destination buffer.
- CRRN2: if no TLV element is found, the UNAVAILABLE_ELEMENT exception is thrown and the current TLV is no longer defined.
- CRRN3: if the method is successful then the corresponding TLV becomes current and dstOffset + length of the copied value is returned.
- CRRN4: The search method is comprehension required flag independent.

6.2.5.11.1.2 Parameter errors

- CRRP1: if dstBuffer is null NullPointerException shall be thrown.
- CRRP2: if dstOffset would cause access outside array bounds ArrayIndexOutOfBoundsException shall be thrown.

6.2.5.11.1.3 Context errors

• CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.

6.2.5.11.2 Test Suite files

Specific triggering: Unrecognized Envelope

Test Script: API_2_ERH_FACYB_BS_1.scr

Test Applet: API_2_ERH_FACYB_BS_1.java

Load Script: API_2_ERH_FACYB_BS_1.ldr

Cleanup Script: API_2_ERH_FACYB_BS_1.clr

Parameter File: API_2_ERH_FACYB_BS_1.par

6.2.5.11.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Initialise the handler	•	•
	findAndCopyValue() with a null dstBuffer	NullPointerException is thrown	
2	appendTLV with TAG: 0D and length 16 Select Text String TLV		
	dstOffset ≥ dstBuffer.length tag = 0Dh dstBuffer.length = 20 dstOffset = 20	ArrayIndexOutOfBoundsException is thrown	
3	<pre>dstOffset < 0 dstBuffer.length = 20 dstOffset = -1</pre>	ArrayIndexOutOfBoundsException is thrown	
4	<pre>dstOffset + length >dstBuffer.length dstBuffer.length = 20 dstOffset = 5</pre>	ArrayIndexOutOfBoundsException is thrown	
5	<pre>length > dstBuffer.length dstBuffer.length = 15 dstOffset = 0</pre>	ArrayIndexOutOfBoundsException is thrown	
6	clear the handler, appendTLV with TAG 02 and Length 02		
	Select a TLV (tag 02h)		
	<pre>findAndCopyValue() tag = 03h</pre>	ToolkitException.UNAVAILABLE_ ELEMENT is thrown	
	Call the getValueLength() method	ToolkitException.UNAVAILABLE_ ELEMENT is thrown.	
7	appendTLV with TAG: 0D and value: 04 00 01 0F		
	Successful call Tag = 0Dh dstBuffer.length = 17 dstOffset = 0	Result of findAndCopyValue() is 17	
8	Compare buffer buffer = 04 00 01 0F	Result is 00h	
9	initialise dstBuffer dstBuffer = 55 55 55		
	Successful call dstBuffer.length = 20 dstOffset = 2	Result of findAndCopyValue() is 19	
10	Compare buffer buffer = 55 55 04 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F 55	Result is 00h	

ld	Description	API Expectation	APDU Expectation
11	clear the handler, appendTLV with TAG: 0D	-	-
	and value: 04 00 01 0F		
	append a 2 nd Text String TLV		
	Successful call	Result of findAndCopyValue() is	
	tag = 0Dh	17	
	dstBuffer.length = 17		
	dstOffset = 0		
12	Compare buffer	Result is 00h	
	buffer = 04 00 01 OF		
13	clear the handler, appendTLV with TAG: 0D		
	and value: 04 00 01 0F		
	Successful call (with tag 8Dh)	Result of findAndCopyValue() is	
	tag = 8Dh	17	
	dstBuffer.length = 17		
	dstOffset = 0		
14	Compare buffer	Result is 00h	
	buffer = 04 00 01 OF		
15	Append tag 0Fh		
	buffer = 00 01 OF		
	Successful call (with tag 8Fh)	Result of findAndCopyValue() is	
	tag = 8Fh	16	
	dstBuffer.length = 16		
	dstOffset = 0		
16	Compare buffer	Result is 00h	
	buffer = 00 01 0F		

6.2.5.11.4 Test Coverage

CRR number	Test case number
N1	8, 10, 12
N2	6
N3	7, 9, 11
N4	13, 14, 15, 16
P1	1
P2	2, 3, 4, 5
C1	Does not apply for EnvelopeResponseHandler

6.2.5.12 Method findAndCopyValue(byte tag, byte occurrence, short valueOffset, byte[] dstBuffer, short dstOffset, short dstLength)

Test Area Reference: API_2_ERH_FACYBBS_BSS

6.2.5.12.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

6.2.5.12.1.1 Normal execution

• CRRN1: looks for the indicated occurrence of a TLV element from the beginning of a TLV list and copy its value into a destination buffer.

- CRRN2: if no TLV element is found, the UNAVAILABLE_ELEMENT exception is thrown and the current TLV is no longer defined.
- CRRN3: if the method is successful then the corresponding TLV becomes current and dstOffset + dstLength is
 returned.
- CRRN4: The search method is comprehension required flag independent.

6.2.5.12.1.2 Parameter errors

- CRRP1: if dstBuffer is null NullPointerException shall be thrown.
- CRRP2: if dstOffset or dstLength or both would cause access outside array bounds, or if dstLength is negative ArrayIndexOutOfBoundsException shall be thrown.
- CRRP3: if valueOffset, dstLength or both are out of the current TLV an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException OUT_OF_TLV_BOUNDARIES.

6.2.5.12.1.3 Context errors

• CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.

6.2.5.12.2 Test Suite files

Specific triggering: Unrecognized Envelope

Test Script: API_2_ERH_FACYBBS_BSS_1.scr

Test Applet: API_2_ERH_FACYBBS_BSS_1.java

Load Script: API_2_ERH_FACYBBS_BSS_1.ldr

Cleanup Script: API_2_ERH_FACYBBS_BSS_1.clr

Parameter File: API_2_ERH_FACYBBS_BSS_1.par

6.2.5.12.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Initialise the handler		
	findAndCopyValue() with a null dstBuffer	NullPointerException is thrown	
2	appendTLV with TAG: 0D and length 16		
	dstOffset ≥ dstBuffer.length	ArrayIndexOutOfBoundsExceptio	
	tag = 0Dh, occurrence = 1	n is thrown	
	valueOffset = 0		
	dstBuffer.length = 5		
	dstOffset = 5		
	dstLength = 1		
3	dstOffset < 0	ArrayIndexOutOfBoundsExceptio	
	dstBuffer.length = 5	n is thrown	
	dstOffset = -1		
	dstLength = 1		
4	dstLength >dstBuffer.length	ArrayIndexOutOfBoundsExceptio	
	dstBuffer.length = 5	n is thrown	
	dstOffset = 0		
	dstLength = 6		
5	dstOffset + dstLength >dstBuffer.length	ArrayIndexOutOfBoundsExceptio	
	dstBuffer.length = 5	n is thrown	
	dstOffset = 3		
	dstLength = 3		
6	dstLength < 0	ArrayIndexOutOfBoundsExceptio	
	dstBuffer.length = 5	n is thrown	
	dstOffset = 0		
	dstLength = -1		

ld	Description	API Expectation	APDU Expectation
7	appendTLV with TAG: 0D and length 6	AiTExpectation	Ai Do Expectation
	valueOffset ≥ Text String Length	ToolkitException.OUT_OF_TLV_	
	tag = 0Dh, occurrence = 1	BOUNDARIES is thrown	
	valueOffset = 6		
	dstBuffer.length = 15		
	dstOffset = 0 dstLength = 1		
8	valueOffset < 0	ToolkitException.OUT_OF_TLV_	
	valueOffset = -1	BOUNDARIES is thrown	
	dstBuffer.length = 15		
	dstOffset = 0		
9	dstLength = 1 dstLength > Text String length	ToolkitException.OUT_OF_TLV_	
9	valueOffset = 0	BOUNDARIES is thrown	
	dstBuffer.length = 15	DOONDAINEO IS UIIOWII	
	dstOffset = 0		
40	dstLength = 7	Tablide Constitution OUT OF TIVE	
10	valueOffset + dstLength > Text String length valueOffset = 2	ToolkitException.OUT_OF_TLV_ BOUNDARIES is thrown	
	dstBuffer.length = 15	BOUNDARIES IS UITOWIT	
	dstOffset = 0		
	dstLength = 5		
11	clear the handler, appendTLV with TAG 02 and		
<u> </u>	Length 02		
	Select a TLV (tag 02h) findAndCopyValue()	Tablide Commission LINIAN (ALL ADLE	
1	tag = ODh	ToolkitException.UNAVAILABLE_ ELEMENT is thrown	
L	occurrence = 2	ELEIVIEIN I 15 (IIIOWII	
	Call the getValueLength() method	ToolkitException.UNAVAILABLE_	
		ELEMENT is thrown.	
12	clear the handler and appendTLV with TAG:		
	0D and value: 04 00 01 0F		
	Successful call	Result of findAndCopyValue() is	
	<pre>tag = 0Dh, occurrence = 1 valueOffset = 0</pre>	17	
	dstBuffer.length = 17		
	dstOffset = 0		
40	dstLength = 17	D 4: 001	
13	Compare buffer buffer = 04 00 01 0F	Result is 00h	
14	initialise dstBuffer		
' '	dstBuffer = 55 55 55		
	Successful call	Result of findAndCopyValue() is	
	tag = 0Dh, occurrence = 1	15	
	<pre>valueOffset = 2 dstBuffer.length = 20</pre>		
	dstBuffer.length = 20 dstOffset = 3		
	dstLength = 12		
15	Compare buffer	Result is 00h	
	buffer =		
	55 55 55 01 02 03 04 05 06 07		
1	08 09 0A 0B 0C		
	55 55 55 55 55		
16	Append a Text String TLV		
	tag = 0D buffer = 00 11 22 33 44 55 (no specific		
	DCS byte)		
	Successful call	Result of findAndCopyValue() is	
	tag = 0Dh, occurrence = 1	17	
	<pre>valueOffset = 0 dstBuffer.length = 20</pre>		
	dstBuller.length = 20 dstOffset = 0		
	dstLength = 17		
17	Compare buffer	Result is 00h	
1.5	buffer = 04 00 01 0F	D # (f 12 10 11 10 11 11 11 11 11 11 11 11 11 11	
18	Successful call tag = 0Dh, occurrence = 2	Result of findAndCopyValue() is 6	
	valueOffset = 0		
	dstBuffer.length = 6		
	dstOffset = 0		
	dstLength = 6		

ld	Description	API Expectation	APDU Expectation
19	Compare buffer	Result is 00h	-
	buffer = 00 11 22 33 44 55		
20	clear the handler and appendTLV with TAG:		
	0D and value: 04 00 01 0F		
	Successful call (with tag 8Dh)	Result of findAndCopyValue () is	
	tag = 8Dh	17	
	occurrence = 1		
	<pre>valueOffset = 0</pre>		
	dstBuffer.length = 17		
	dstOffset = 0		
	dstLength = 17		
21	Compare buffer	Result is 00h	
	buffer = 04 00 01 0F		
22	Append tag 0Fh		
	buffer = 00 01 0F		
	Successful call (with tag 8Fh)	Result of findAndCopyValue () is	
	tag = 8Fh	16	
	occurrence = 1		
	valueOffset = 0		
	dstBuffer.length = 16		
	dstOffset = 0		
	dstLength = 16		
23	Compare buffer	Result is 00h	
	buffer = 00 01 0F		
24	Successful call, findAndCopyValue with length	Result of findAndCopyValue () is	
	=0	16	
	dstBuffer.length = 16		
	dstOffset = 16		
	dstLength = 0		

6.2.5.12.4 Test Coverage

CRR number	Test case number
N1	13, 15, 17, 19
N2	11
N3	12, 14, 16, 18, 24
N4	20, 21, 22, 23
P1	1
P2	2, 3, 4, 5, 6
P3	7, 8, 9, 10
C1	Does not apply for EnvelopeResponseHandler

6.2.5.13 Method findAndCompareValue(byte tag, byte[] compareBuffer, short compareOffset)

Test Area Reference: API_2_ERH_FACRB_BS

6.2.5.13.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

6.2.5.13.1.1 Normal execution

Looks for the first occurrence of a TLV element from beginning of a TLV list and compare its value with a buffer:

• CRRN1: if no TLV element is found, the UNAVAILABLE_ELEMENT exception is thrown and the current TLV is no longer defined.

- CRRN2: if the method is successful then the corresponding TLV becomes current.
- CRRN3: if identical returns 0.
- CRRN4: if the first miscomparing byte in simple TLV is less than that in compareBuffer returns -1.
- CRRN5: if the first miscomparing byte in simple TLV is greater than that in compareBuffer returns 1.
- CRRN6: The search method is comprehension required flag independent.

6.2.5.13.1.2 Parameter errors

- CRRP1: if compareBuffer is null NullPointerException shall be thrown.
- CRRP2: if compareOffset would cause access outside array bounds ArrayIndexOutOfBoundsException shall be thrown.

6.2.5.13.1.3 Context errors

• CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.

6.2.5.13.2 Test Suite files

Specific triggering: Unrecognized Envelope

Test Script: API_2_ERH_FACRB_BS_1.scr

Test Applet: API_2_ERH_FACRB_BS_1.java

Load Script: API_2_ERH_FACRB_BS_1.ldr

Cleanup Script: API_2_ERH_FACRB_BS_1.clr

Parameter File: API_2_ERH_FACRB_BS_1.par

6.2.5.13.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	appendTLV with TAG: 0D and length 16		-
	findAndCompareValue() with a null dstBuffer and tag 0Dh	NullPointerException is thrown	
2			
	<pre>compareOffset ≥ compareBuffer.length tag = 0Dh compareBuffer.length = 20 compareOffset = 20</pre>	ArrayIndexOutOfBoundsException is thrown	
3	<pre>compareOffset < 0 compareBuffer.length = 20 compareOffset = -1</pre>	ArrayIndexOutOfBoundsException is thrown	
4	<pre>compareOffset + length ></pre>	ArrayIndexOutOfBoundsException is thrown	
5	<pre>length > compareBuffer.length compareBuffer.length = 15 compareOffset = 0</pre>	ArrayIndexOutOfBoundsExceptio n is thrown	
6	clear the handler, appendTLV with TAG 02 and Length 02		
	Select a TLV (tag 02h)		
	<pre>findAndCompareValue() tag = 03h</pre>	ToolkitException.UNAVAILABLE_ ELEMENT is thrown	
7	Verify current TLV getValueLength()	ToolkitException.UNAVAILABLE_ ELEMENT is thrown.	

ld	Description	API Expectation	APDU Expectation
	•	AFI Expectation	APDO Expectation
8	clear the handler and appendTLV with TAG:		
	0D and value: 04 00 01 0F		
	Initialise compareBuffer		
	compareBuffer = 04 00 01 OF		
	Compare buffers	Result is 00h	
	tag = 0Dh	Result is oon	
	compareOffset = 0		
9	Verify current TLV	Result is 17	
	getValueLength()		
10	Initialise compareBuffer		
	compareBuffer =		
	04 00 01 10		
	Compare buffers with same parameters	Result is -1	
11	Initialise compareBuffer		
	compareBuffer =		
	03 00 01 0F		
	Compare buffers with same parameters	Result is +1	
12	Initialise compareBuffer		
	compareBuffer = 55 55 04 00 01		
	02 03 04 05 06		
	07 08 09 0A 0B		
	OC OD OE OF 55		
	Compare buffers	Result is 00h	
	compareOffset = 2		
13	append a Text String TLV		
	tag = 0Dh		
	buffer = 00 11 22 33 44 55		
	Initialise compareBuffer		
	compareBuffer = 55 55 04 00 01		
	02 03 04 05 06		
	07 08 09 0A 0B		
	OC OD OE OF 55		
	Compare buffers	Result is 00h	
	compareOffset = 2		
14	Initialise compareBuffer		
	compareBuffer =		
	55 55 04 01 01		
	02 03 04 05 06 07 08 09 0A 0B		
	OC OD OE OF 55		
	Compare buffers	Result is -1	
	compareOffset = 2		
15	Initialise compareBuffer		
. •	compareBuffer =		
	55 55 04 00 01		
	02 03 04 05 06		
	07 08 09 0A 0B		
	0C 0D 0D 10 55	Popult in 11	
	Compare buffers compareOffset = 2	Result is +1	
16	clear the handler and appendTLV with TAG:		
'0	0D and value: 04 00 01 0F		
	Initialise compareBuffer		
	compareBuffer = 04 00 01 0F		
	Successful call (with tag 8Dh)	Result is 00h	1
	tag = 8Dh	1. Codit to Con	
	compareBuffer.length = 17		
	compareOffset = 0		
17	Append tag 0Fh		
	buffer = 00 01 0F		
	Initialise compareBuffer		
	compareBuffer = 00 01 0F		
	Successful call (with tag 8Fh)	Result is 00h	
	tag = 8Fh		
	<pre>compareBuffer.length = 16 compareOffset = 0</pre>		
	oomparcorrace - o		I

6.2.5.13.4 Test Coverage

CRR number	Test case number
N1	6,7
N2	7,9
N3	8, 13, 12
N4	10, 14
N5	11, 15
N6	17, 16
P1	1
P2	2, 3, 4, 5
C1	Does not apply for Envelope response handler

6.2.5.14 Method findAndCompareValue(byte tag, byte occurrence, short valueOffset, byte[] compareBuffer, short compareOffset, short compareLength)

Test Area Reference: API_2_ERH_FACRBBS_BSS

6.2.5.14.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

6.2.5.14.1.1 Normal execution

Looks for the indicated occurrence of a TLV element from the beginning of a TLV list and compare its value with a buffer:

- CRRN1: if no TLV element is found, the UNAVAILABLE_ELEMENT exception is thrown and the current TLV is no longer defined.
- CRRN2: if the method is successful then the corresponding TLV becomes current.
- CRRN3: if identical 0 is returned.
- CRRN4: if the first miscomparing byte in simple TLV is less than that in compareBuffer -1 is returned.
- CRRN5: if the first miscomparing byte in simple TLV is greater than that in compareBuffer 1 is returned
- CRRN6: The search method is comprehension required flag independent.

6.2.5.14.1.2 Parameter errors

- $\bullet \quad CRRP1: if \ compare Buffer \ is \ null \ Null Pointer Exception \ shall \ be \ thrown.$
- CRRP2: if compareOffset or compareLength or both would cause access outside array bounds, or if compareLength is negative ArrayIndexOutOfBoundsException shall be thrown.
- CRRP3: if valueOffset, compareLength or both are out of the current TLV an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException OUT_OF_TLV_BOUNDARIES.
- CRRP4: if an input parameter is not valid (e.g. occurrence = 0) an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException BAD_INPUT_PARAMETER.

6.2.5.14.1.3 Context errors

• CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.

6.2.5.14.2 Test Suite files

Specific triggering: Unrecognized Envelope

Test Script: API_2_ERH_FACRBBS_BSS_1.scr

Test Applet: API_2_ERH_FACRBBS_BSS_1.java

Load Script: API_2_ERH_FACRBBS_BSS_1.ldr

Cleanup Script: API_2_ERH_FACRBBS_BSS_1.clr

Parameter File: API_2_ERH_FACRBBS_BSS_1.par

6.2.5.14.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Initialise the handler	Ari Expectation	AF DO Expectation
	findAndCompareValue() with a null	Null Dointer Eveention is through	
	compareBuffer	NullPointerException is thrown	
2	clear the handler and appendTLV with TAG:		
2	0D and value: 04 00 01 0F		
		A manufactor to the country of Europe ties	
	compareOffset ≥ compareBuffer.length	ArrayIndexOutOfBoundsExceptio	
	<pre>tag = 0Dh, occurrence = 1 valueOffset = 0</pre>	n is thrown	
	compareBuffer.length = 5		
	compareOffset = 5		
	compareLength = 1		
3	compareOffset < 0	ArrayIndexOutOfBoundsExceptio	
	compareBuffer.length = 5	n is thrown	
	compareOffset = -1		
	compareLength = 1		
4	compareLength >compareBuffer.length	ArrayIndexOutOfBoundsExceptio	
	compareBuffer.length = 5	n is thrown	
	<pre>compareOffset = 0 compareLength = 6</pre>		
5	compareOffset + compareLength	ArrayIndexOutOfBoundsExceptio	
٦	>compareBuffer.length	n is thrown	
	compareBuffer.length = 5	III IS UIIOWII	
	compareOffset = 3		
	compareLength = 3		
6	compareLength < 0	ArrayIndexOutOfBoundsExceptio	
	compareBuffer.length = 5	n is thrown	
	compareOffset = 0		
	compareLength = -1		
	The state of the s		
7	clear the handler and appendTLV with TAG		
	and length of 6	To all the control of the	
	valueOffset ≥ Text String Length	ToolkitException.OUT_OF_TLV_ BOUNDARIES is thrown	
	<pre>tag = 0Dh, occurrence = 1 valueOffset = 6</pre>	BOUNDARIES IS INFOWN	
	compareBuffer.length = 15		
	compareOffset = 0		
	compareLength = 1		
8	valueOffset < 0	ToolkitException.OUT_OF_TLV_	
	valueOffset = -1	BOUNDARIES is thrown	
	compareBuffer.length = 15		
	compareOffset = 0		
9	compareLength = 1 compareLength > Text String length	ToolkitException.OUT_OF_TLV_	
9	valueOffset = 0	BOUNDARIES is thrown	
	compareBuffer.length = 15	DOUNDARIES IS IIIIUWII	
	compareOffset = 0		
	compareLength = 7		
	<u> </u>		

ld	Description	API Expectation	APDU Expectation
10	valueOffset + compareLength > Text String	ToolkitException.OUT_OF_TLV_	.
	length	BOUNDARIES is thrown	
	valueOffset = 2		
	compareBuffer.length = 15		
	compareOffset = 0		
4.	compareLength = 5	T 11/15 (
11	Invalid parameter	ToolkitException.BAD_INPUT_PA	
40	occurrence = 0	RAMETER is thrown	
12	appendTLV with TAG 02 and length 02		
	Select a TLV (tag 02h)		
	findAndCompareValue()	ToolkitException.UNAVAILABLE_	
	tag = 0Dh occurrence = 2	ELEMENT is thrown	
13	Verify current TLV	ToolkitException.UNAVAILABLE_	
13	getValueLength()	ELEMENT is thrown.	
14	clear the handler and appendTLV with TAG:	ELLIVILIAT IS UIIOWII.	
14	0D and value: 04 00 01 0F		
	Initialise compareBuffer		
	compareBuffer =		
	04 00 01 OF		
	findAndCompareValue()	Result is 00h	
	tag = 0Dh, occurrence = 1		
	valueOffset = 0		
	compareOffset = 0		
	compareLength = 17	Decide: 47	
15	Verify current TLV	Result is 17	
4.0	getValueLength()	+	
16	Initialise compareBuffer		
	compareBuffer = 04 00 01 10		
	Compare buffers with same parameters	Result is -1	
17	Initialise compareBuffer	recount to - t	_
' /	compareBuffer =		
	03 00 01 OF		
	Compare buffers with same parameters	Result is +1	
18	Initialise compareBuffer	-	
. 5	compareBuffer =		
	55 55 55 01 02		
	03 04 05 06 07		
	08 09 0A 0B 0C 55 55 55 55 55		
—		Result is 00h	
	Compare buffers valueOffset = 2	IVESUIT IS ON!	
	compareOffset = 3		
L	compareLength = 12		
19	Initialise compareBuffer		
	compareBuffer =		
	55 55 55 02 01		
	03 04 05 06 07		
	08 09 0A 0B 0C		
	55 55 55 55 55 55 Compare buffers with same parameters	Result is -1	
20		Result is -1	
20	Initialise compareBuffer compareBuffer =		
	CompareBuiler		
	03 04 05 06 07		
	08 09 0A 0A 0D		
	55 55 55 55 55		
	Compare buffers with same parameters	Result is +1	
21	append a Text String TLV		
	tag = 0Dh		
	buffer = 00 11 22 33 44 55	_	
	Initialise compareBuffer		
	compareBuffer = 04 00 01 0F		
	findAndCompareValue()	Result is 00h	
	tag = 0Dh, occurrence = 1	IVESUIT IS ON!	
	<pre>tag = UDn, occurrence = 1 valueOffset = 0</pre>		
	compareOffset = 0		
L	compareLength = 17		

ld	Description	API Expectation	APDU Expectation
22	Initialise compareBuffer	_	_
	compareBuffer =		
	00 11 22 33 44 55		
	findAndCompareValue()	Result is 00h	
	tag = 0Dh, occurrence = 2		
	valueOffset = 0		
	<pre>compareOffset = 0 compareLength = 6</pre>		
23	Initialise compareBuffer		
23	compareBuffer =		
	00 11 22 33 44 66		
	findAndCompareValue()	Result is -1	
	tag = 0Dh, occurrence = 2	. toodiit io	
	valueOffset = 0		
	compareOffset = 0		
	compareLength = 6		
24	clear the handler and appendTLV with TAG:		
	0D and value: 04 00 01 0F		
	Initialise compareBuffer		
	compareBuffer = 04 00 01 OF		
	Successful call (with tag 8Dh)	Result is 00h	
	tag = 8Dh, occurrence = 1		
	valueOffset = 0		
	<pre>compareBuffer.length = 17 compareOffset = 0</pre>		
	compareLength = 17		
25	Append tag 0Fh		
25	buffer = 00 01 0F		
	Initialise compareBuffer		
	compareBuffer = 00 01 0F		
	Successful call (with tag 8Fh)	Result is 00h	
	tag = 8Fh, occurrence = 1		
	<pre>valueOffset = 0</pre>		
	compareBuffer.length = 16		
	compareOffset = 0		
	compareLength = 16	D 1/ (C 1A 1O)/ 1	
26	Successful call, findAndCompareValue with	Result of findAndCompareValue	
	length =0	() is 00	
	CompareBuffer.length = 16		
	compareOffset = 16		
	compareLength = 0		

6.2.5.14.4 Test Coverage

CRR number	Test case number
N1	12,13
N2	15,13
N3	14, 18, 22, 21, 26
N4	16, 19, 23
N5	17, 19
N6	25, 24
P1	1
P2	2, 3, 4, 5, 6
P3	7, 8, 9, 10
P4	11
C1	Does not apply for EnvelopeResponseHandler

6.2.5.15 Method appendArray

Test Area Reference: API_2_ERH_APDA_BSS

6.2.5.15.1 Conformance requirement:

The method with following header shall be compliant to its definition in the API.

6.2.5.15.1.1 Normal execution

- CRRN1: appends a buffer into the EditHandler buffer
- CRRN2: a successful append does not modify the TLV selected

6.2.5.15.1.2 Parameters error

- CRRP1: if buffer is null, a java.lang.NullPointerException is thrown
- CRRP2: if offset or length or both would cause access outside the array bounds, or if length is negative, a java.lang.ArrayIndexOutOfBoundsException is thrown.

6.2.5.15.1.3 Context errors

- CRRC1: if the EditHandler buffer is too small to append the requested data, a ToolkitException is thrown with reason code HANDLER_OVERFLOW
- CRRC2: if the EditHandler buffer is busy, a ToolkitException is thrown with reason code HANDLER_NOT_AVAILABLE

6.2.5.15.2 Test suite files

Specific triggering: Unrecognized Envelope

Test Script: API_2_ERH_APDA_BSS_1.scr
Test Applet: API_2_ERH_APDA_BSS_1.java
Load Script: API_2_ERH_APDA_BSS_1.ldr
Cleanup Script: API_2_ERH_APDA_BSS_1.clr
Parameter File: API_2_ERH_APDA_BSS_1.par

6.2.5.15.3 Test procedure

ld	Description	API Expectation	APDU Expectation
	Initialise the envelope response handler with a TLV of length 1		
1	Null buffer	NullPointerException is thrown	
2	offset ≥ buffer.length	ArrayIndexOutOfBoundsExceptio	
	buffer.length = 5	n is thrown	
	offset = 5		
	length = 1		
3	offset < 0	ArrayIndexOutOfBoundsExceptio	
	buffer.length = 5	n is thrown	
	offset = -1		
	length = 1		
4	length > buffer.length	ArrayIndexOutOfBoundsExceptio	
	buffer.length = 5	n is thrown	
	offset = 0		
	length = 6		
5	offset + length > buffer.length	ArrayIndexOutOfBoundsExceptio	
	buffer.length = 5	n is thrown	
	offset = 3		
	length = 3		

ld	Description	API Expectation	APDU Expectation
6	length < 0	ArrayIndexOutOfBoundsExceptio	-
	buffer.length = 5	n is thrown	
	offset = 0		
	length = -1		
7	Handler overflow	ToolkitException.HANDLER_OVE	
	buffer.length = 256	RFLOW is thrown	
	offset = 0		
	length = 256		
8	append the handler with TLVs:		
	81 03 11 22 33		
	82 02 99 77		
	findTLV 0x81		
	Successful call		
	buffer = FF FE F8		
	offset = 0		
	length = 8		
	Verify Current TLV: Call getValueLength()	Result is 03h	
9	Clear the handler		
	Successful call		
	buffer = FF FE F8		
	offset = 0		
	length = 8		
	Call copy() method		
	Compare handler	Result is 00h	
	compareBuffer = FF FE F8		
10	Successful call		
	buffer = 00 01 07		
	offset = 2		
	length = 6		
	Call copy() method		
	Compare handler	Result is 00h	
	compareBuffer = FF FE F8 02 03 07		
11	Successful call		
	buffer = 11 22 88		
	offset = 2		
	length = 4		
	Call copy() method		
	Compare handler	Result is 00h	
	compareBuffer = FF FE F8 02 03 07 33		
	44 55 66		

6.2.5.15.4 Test Coverage

CRR number	Test case number
N1	9, 10, 11
N2	8
N3	
P1	1
P2	2, 3, 4, 5, 6
C1	7
C2	Does not apply for EnvelopeResponseHandler

6.2.5.16 Method appendTLV(byte tag, byte value)

Test Area Reference: API_2_ERH_APTLBB

6.2.5.16.1 Conformance requirement:

The method with following header shall be compliant to its definition in the API.

6.2.5.16.1.1 Normal execution

- CRRN1: Appends a TLV element to the current TLV list (1-byte element).
- CRRN2: A successful append does not modify the TLV selected.

6.2.5.16.1.2 Parameters error

No requirements

6.2.5.16.1.3 Context errors

- CRRC1: if the EditHandler buffer is too small to append the requested data, a ToolkitException is thrown with reason code HANDLER_OVERFLOW
- CRRC2: if the EditHandler buffer is busy, a ToolkitException is thrown with reason code HANDLER_NOT_AVAILABLE

6.2.5.16.2 Test suite files

Specific triggering: Unrecognized Envelope

Test Script: API_2_ERH_APTLBB_1.scr

Test Applet: API_2_ERH_APTLBB_1.java

Load Script: API_2_ERH_APTLBB_1.ldr

Cleanup Script: API_2_ERH_APTLBB_1.clr

Parameter File: API_2_ERH_APTLBB_1.par

6.2.5.16.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Call appendArray()		
ļ .	length = 253		
	Handler Overflow: Call twice the	ToolkitException.HANDLER_OVE	
	appendTLV()method	RFLOW is thrown by one of the	
	V · · ·	two.	
2	append the handler with TLVs:		
	81 03 11 22 33		
	82 02 99 77		
	Select Command Details TLV		
	Call the appendTLV() method		
	Verify Current TLV: Call getValueLength()	Result is 03h	
3	Clear the handler		
	Successful call		
	tag = 84h		
	value = 00h		
	Call copy() method		
	Compare handler	Result is 00h	
	compareBuffer = 84 01 00		
4	Successful call		
	tag = 01h		
	value = FEh		
	Call copy() method		
	Compare handler	Result is 00h	·
	compareBuffer = 84 01 00 01 01 FE		

6.2.5.16.4 Test Coverage

CRR number	Test case number
N1	3, 4
N2	2
C1	1
C2	Does not apply for EnvelopeResponseHandler

6.2.5.17 Method appendTLV(byte tag, byte value1, byte value2)

Test Area Reference: API_2_ERH_APTLBBB

6.2.5.17.1 Conformance requirements:

The method with following header shall be compliant to its definition in the API.

6.2.5.17.1.1 Normal execution

- CRRN1: Appends a TLV element to the current TLV list (2-byte element).
- CRRN2: A successful append does not modify the TLV selected.

6.2.5.17.1.2 Parameters error

No requirements

6.2.5.17.1.3 Context errors

- CRRC1: if the EditHandler buffer is too small to append the requested data, a ToolkitException is thrown with reason code HANDLER_OVERFLOW
- CRRC2: if the EditHandler buffer is busy, a ToolkitException is thrown with reason code HANDLER_NOT_AVAILABLE

6.2.5.17.2 Test suite files

Specific triggering: Unrecognized Envelope

Test Script: API_2_ERH_APTLBBB_1.scr

Test Applet: API_2_ERH_APTL BBB_1.java

Load Script: API_2_ERH_APTL BBB_1.ldr

Cleanup Script: API_2_ERH_APTLBBB_1.clr

Parameter File: API_2_ERH_APTLBBB_1.par

6.2.5.17.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Call the appendArray with length of 253		
	Handler Overflow: Call the appendTLV()	ToolkitException.HANDLER_OVE	
	method	RFLOW is thrown	

ld	Description	API Expectation	APDU Expectation
2	clear the handler, append the handler with	_	_
	TLVs:		
	81 03 11 22 33		
	82 02 99 77		
	Select Command Details TLV		
	Call the appendTLV() method		
	Verify Current TLV: Call getValueLength()	Result is 03h	
3	Clear the handler		
	Successful call		
	tag = 84h		
	value1 = 00h		
	value2 = 01h		
	Call copy() method		
	Compare handler	Result is 00h	
	compareBuffer = 84 02 00 01		
4	Successful call		
	tag = 01h		
	value1 = FEh		
	value2 = FDh		
	Call copy() method		
	Compare handler	Result is 00h	
	compareBuffer = 84 02 00 01 01 02 FE FD		

6.2.5.17.4 Test Coverage

CRR number	Test case number
N1	3, 4
N2	2
C1	1
C2	Does not apply for EnvelopeResponseHandler

6.2.5.18 Method appendTLV(byte tag, byte[] value, short valueoffset, short valuelength)

Test Area Reference: API_2_ERH_APTLB_BSS

6.2.5.18.1 Conformance requirement:

The method with following header shall be compliant to its definition in the API.

6.2.5.18.1.1 Normal execution

- CRRN1: Appends a TLV element to the current TLV list (byte-array element).
- CRRN2: A successful append does not modify the TLV selected.

6.2.5.18.1.2 Parameters error

- CRRP1: if value is null, a java.lang.NullPointerException is thrown
- CRRP2: if valueoffset or valuelength or both would cause access outside the array bounds, or if length is negative, a java.lang.ArrayIndexOutOfBoundsException is thrown.

6.2.5.18.1.3 Context errors

- CRRC1: if the EditHandler buffer is too small to append the requested data, a ToolkitException is thrown with reason code HANDLER_OVERFLOW
- CRRC2: if the EditHandler buffer is busy, a ToolkitException is thrown with reason code HANDLER_NOT_AVAILABLE
- CRRC3: if valuelength is greater than 255, a ToolkitException is thrown with reason code BAD_INPUT_PARAMETER

6.2.5.18.2 Test suite files

Specific triggering: Unrecognized Envelope

Test Script: API_2_ERH_APTLB_BSS_1.scr

Test Applet: API_2_ERH_APTLB_BSS_1.java

Load Script: API_2_ERH_APTLB_BSS_1.ldr

Cleanup Script: API_2_ERH_APTLB_BSS_1.clr

Parameter File: API_2_ERH_APTLB_BSS_1.par

6.2.5.18.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Null value	NullPointerException is thrown	-
2	valueOffset ≥ value.length	ArrayIndexOutOfBoundsExceptio	
	value.length = 5	n is thrown	
	valueOffset = 5		
	valueLength = 1		
3	valueOffset < 0	ArrayIndexOutOfBoundsExceptio	
	value.length = 5	n is thrown	
	valueOffset = -1		
L.	valueLength = 1	1 1 0 10/0	
4	valueLength > value.length	ArrayIndexOutOfBoundsExceptio	
	<pre>value.length = 5 valueOffset = 0</pre>	n is thrown	
	valueLength = 6		
5	ValueOffset + valueLength > value.length	ArrayIndexOutOfBoundsExceptio	
٥	value.length = 5	n is thrown	
	valueOffset = 3	III IS UIIOWII	
	valueLength = 3		
6	valueLength < 0	ArrayIndexOutOfBoundsExceptio	
	value.length = 5	n is thrown	
	valueOffset = 0		
	valueLength = -1		
7	Handler overflow	ToolkitException.HANDLER_OVE	
	value.length = 254	RFLOW is thrown	
	valueOffset = 0		
_	valueLength = 254	Tablide DAD INDUT DA	
8	Bad parameter	ToolkitException.BAD_INPUT_PA	
	<pre>value.length = 256 valueOffset = 0</pre>	RAMETER is thrown	
	valueLength = 256		
9	clear the handler, append the handler with		
	TI Vs:		
	81 03 11 22 33		
	82 02 99 77		
	Select Command Details TLV		
	Successful call		
	tag = 04		
	value = FF FE F8		
	<pre>valueOffset = 0</pre>		
	valueLength = 8		
	Verify Current TLV: Call getValueLength()	Result is 03h	

ld	Description	API Expectation	APDU Expectation
10	Clear the handler		
	Successful call		
	tag = 04		
	value = FF FE F8		
	valueOffset = 0		
	valueLength = 8		
	Call copy() method		
	Compare handler	Result is 00	
	CompareBuffer = 04 08 FF FE F8		
11	Successful call		
	tag = 85h		
	value = 00 01 07		
	valueOffset = 2		
	valueLength = 6		
	Call copy() method		
	Compare handler	Result is 00	
	compareBuffer = 04 08 FF FE F8 85 06 02		
	03 07		
12	Successful call		
	tag = 01		
	value = 11 22 88 valueOffset = 2		
	valueLength = 4		
	Call copy() method		
		Danutia 00	
	Compare handler compareBuffer = 04 08 FF FE F8 85 06 02	Result is 00	
	03 07 01 04 33 44 55 66		
13	Clear the handler		
-10	Successful call		
	tag = 04		
	value = 00 01 7F		
	valueOffset = 0		
	valueLength = 80h		
	Call copy() method		
	Compare handler	Result is 00	
	compareBuffer = 04 81 80 00 017F		

6.2.5.18.4 Test Coverage

CRR number	Test case number
N1	10, 11, 12, 13
N2	9
P1	1
P2	2, 3, 4, 5, 6
C1	7
C2	Does not apply for EnvelopeResponseHandler
C3	8

6.2.5.19 Method appendTLV(byte tag, byte value1, byte[] value2, short value2offset, short value2length)

Test Area Reference: API_2_ERH_APTLBB_BSS

6.2.5.19.1 Conformance requirement:

The method with following header shall be compliant to its definition in the API.

6.2.5.19.1.1 Normal execution

- CRRN1: Appends a TLV element to the current TLV list (1 byte and a byte-array element).
- CRRN2: A successful append does not modify the TLV selected.

6.2.5.19.1.2 Parameters error

- CRRP1: if value2 is null, a java.lang.NullPointerException is thrown
- CRRP2: if value2offset or value2length or both would cause access outside the array bounds, or if length is negative, a java.lang.ArrayIndexOutOfBoundsException is thrown.

6.2.5.19.1.3 Context errors

- CRRC1: if the EditHandler buffer is too small to append the requested data, a ToolkitException is thrown with reason code HANDLER_OVERFLOW
- CRRC2: if the EditHandler buffer is busy, a ToolkitException is thrown with reason code HANDLER_NOT_AVAILABLE
- CRRC3: if valuelength is greater than 255, a ToolkitException is thrown with reason code BAD_INPUT_PARAMETER

6.2.5.19.2 Test suite files

Specific triggering: Unrecognized Envelope

Test Script: API_2_ERH_APTLBB_BSS_1.scr

Test Applet: API_2_ERH_APTLBB_BSS_1.java

Load Script: API_2_ERH_APTLBB_BSS_1.ldr

Cleanup Script: API_2_ERH_APTLBB_BSS_1.clr

Parameter File: API_2_ERH_APTLBB_BSS_1.par

6.2.5.19.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Null value2	NullPointerException is thrown	•
2	value2Offset ≥ value2.length	ArrayIndexOutOfBoundsExceptio	
	value2.length = 5	n is thrown	
	value20ffset = 5		
	value2Length = 1		
3	value2Offset < 0	ArrayIndexOutOfBoundsExceptio	
	value2.length = 5	n is thrown	
	value20ffset = -1		
	value2Length = 1		
4	value2Length > value2.length	ArrayIndexOutOfBoundsExceptio	
	value2.length = 5	n is thrown	
	value2Offset = 0		
_	value2Length = 6	A 1 1 0 10'D 1 5	
5	value2Offset + value2Length > value2.length	ArrayIndexOutOfBoundsExceptio	
	<pre>value2.length = 5 value20ffset = 3</pre>	n is thrown	
	value2Driset = 3 value2Length = 3		
6	value2Length < 0	ArrayIndexOutOfBoundsExceptio	
O	value2.length = 5	n is thrown	
	value20ffset = 0	II IS UIIOWII	
	value2Length = -1		
7	Handler overflow	ToolkitException.HANDLER_OVE	
	value2.length = 254	RFLOW is thrown	
	value20ffset = 0		
	value2Length = 254		

ld	Description	API Expectation	APDU Expectation
8	Bad parameter	ToolkitException.BAD_INPUT_PA	•
	value2.length = 256	RAMETER is thrown	
	value20ffset = 0		
	value2Length = 256		
9	clear the handler, append the handler with		
	TLVs:		
	81 03 11 22 33 82 02 99 77		
	Select Command Details TLV		
	Successful call		
	tag = 04		
	value1 = 05		
	value2 = FF FE F8		
	<pre>value2Offset = 0 value2Length = 8</pre>		
	Verify Current TLV: Call getValueLength()	Result is 03h	
10	Clear the handler	Result is USIT	
10	Successful call		
	tag = 04		
	value1 = 05		
	value2 = FF FE F8		
	value2Offset = 0		
	value2Length = 8		
	Call copy() method	Describie 00	
	Compare handler CompareBuffer = 04 09 05 FF FE F8	Result is 00	
11	Successful call		
	tag = 85h		
	value1 = 55h		
	value2 = 00 01 07		
	value20ffset = 2		
	value2Length = 6 Call copy() method		
	Compare handler	Result is 00	
	compareBuffer =	Result is 00	
	04 09 05 FF FE F8		
	85 07 55 02 03 07		
12	Successful call		
	tag = 01 value1 = 44h		
	value1 = 44n value2 = 11 22 88		
	value2Offset = 2		
	value2Length = 4		
	Call copy() method		
	Compare handler	Result is 00	
	CompareBuffer = 04 09 05 FF FE F8		
	04 09 05 FF FE F8 85 07 55 02 03 07		
	01 05 44 33 44 55 66		
13	Clear the handler		
	Successful call		
	tag = 04		
	value1 = 00		
	value2 = 01 7F value20ffset = 0		
	value2Driset = 0 value2Length = 7Fh		
	Call copy() method		
	Compare handler	Result is 00	
	compareBuffer = 04 81 80 00 017F		

6.2.5.19.4 Test Coverage

CRR number	Test case number
N1	10, 11, 12, 13
N2	9
P1	1
P2	2, 3, 4, 5, 6
C1	7
C2	Does not apply for EnvelopeResponseHandler
C3	8

6.2.5.20 Method clear

Test Area Reference: API_2_ERH_CLER

6.2.5.20.1 Conformance requirement:

The method with following header shall be compliant to its definition in the API.

6.2.5.20.1.1 Normal execution

• CRRN1: Clears the TLV list of an EditHandler and resets the current TLV selected.

6.2.5.20.1.2 Parameters error

No requirements

6.2.5.20.1.3 Context errors

• CRRC1: if the EditHandler buffer is busy, a ToolkitException is thrown with reason code HANDLER_NOT_AVAILABLE

6.2.5.20.2 Test suite files

Specific triggering: Unrecognized Envelope

Test Script: API_2_ERH_CLER_1.scr

Test Applet: API_2_ERH_CLER_1.java

Load Script: API_2_ERH_CLER_1.ldr

Cleanup Script: API_2_ERH_CLER_1.clr

Parameter File: API_2_ERH_CLER_1.par

6.2.5.20.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	append the handler with TLVs:	Result of getLength() is not null	
	81 03 11 22 33		
	82 02 99 77		
	Select Command Details TLV		
	Call the getLength() method		
	Clear the handler	Result of getLength() is 0	
	Call the getLength() method		
2	Call the getValueLength() method	ToolkitException.UNAVAILABLE_	
		ELEMENT is thrown	

6.2.5.20.4 Test Coverage

CRR number	Test case number	
N1	1, 2	
C1	Does not apply for EnvelopeResponseHandler	

6.2.5.21 Method getCapacity

Test Area Reference: API_2_ERH_GCAP

6.2.5.21.1 Conformance Requirement:

The method with following header shall be compliant to its definition in the API.

Public byte getCapacity()

6.2.5.21.1.1 Normal execution

• CRRN1: The method shall return the maximum size of the Simple TLV list managed by the handler.

6.2.5.21.1.2 Context errors

• CRRC1: The method shall throw HANDLER_NOT_AVAILABLE ToolkitException if the handler is busy.

6.2.5.21.2 Test suite files

Test Script: API_2_ERH_GCAP_1.scr

Test Applet: API_2_ERH_GCAP_1.java

Load Script: API_2_ERH_GCAP_1.ldr

Cleanup Script: API_2_ERH_GCAP_1.clr

Parameter File: API_2_ERH_GCAP_1.par

ld	Description	API Expectation	APDU Expectation
1	EnvelopeResponseHandler available		
	1- Send envelope SMS-PP Formatted 2- The applet calls getTheHandler() method 3- The applet calls getCapacity() method on the EnvelopeResponseHandler	1- Applet is triggered2- No exception is thrown3- No exception is thrown	
	4- The applet fills the handler with the maximum capacity using AppendTLV() method 5- The applet calls clear() method on the	4- No exception is thrown 5- No exception is thrown	
	EnvelopeResponseHandler 6- The applet fills the handler with the maximum capacity plus one, using AppendTLV() method	6- HANDLER_OVERFLOW exception is thrown	

6.2.5.21.4 Test Coverage

CRR number	Test case number
N1	1
C1	Tested in Framework
	part: FWK_MHA_ERHD

6.2.6 Class MEProfile

6.2.6.1 Method check (byte index)

Test Area Reference: API_2_MEP_CHECB

6.2.6.1.1 Conformance requirement:

The method with following header shall compliant to its definition in the API.

6.2.6.1.1.1 Normal execution

• CRRN1: The method checks a facility in the handset profile: returns true if supported and false otherwise.

6.2.6.1.1.2 Parameters error

No requirements.

6.2.6.1.1.3 Context errors

 CRRC1: The method shall throw ME_PROFILE_NOT_AVAILABLE ToolkitException if Terminal Profile data are not available

6.2.6.1.2 Test suite files

Specific triggering:

EVENT_STATUS_COMMAND

No Additional requirements for the GSM personalization:

Test Script: API_2_MEP_CHECB_1.scr

Test Applet: API_2_MEP_CHECB_1.java

Load Script: API_2_MEP_CHECB_1.ldr

Cleanup Script: API_2_MEP_CHECB_1.clr

Parameter File: API_2_MEP_CHECB_1.par

6.2.6.1.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	No Terminal Profile is registered	ME_PROFILE_NOT_AVAILABLE	
	Triggered by status command	ToolkitException is thrown	
	Index = 1	•	
2	Terminal Profile, Facility is supported	true is returned by the method	
	index = 0	_	
3	Terminal Profile, Facility is not supported	false is returned by the method	
	index = 15		

6.2.6.1.4 Test Coverage

CRR number	Test case number
N1	2,3
C1	1

6.2.6.2 Method check (byte [] mask, short offset, short length)

Test Area Reference: API_2_MEP_CHEC_BSS

6.2.6.2.1 Conformance requirement:

The method with following header shall compliant to its definition in the API.

6.2.6.2.1.1 Normal execution

- CRRN1: The method checks all the facilities corresponding to bits set to 1 in the mask buffer: returns true if they are all supported and false if not.
- CRRN2: The method returns true if the length to check is 0.

6.2.6.2.1.2 Parameters error

- CRRP1: The method shall throw java.lang.NullPointerException if mask is null.
- CRRP2: The method shall throw java.lang.ArrayIndexOutOfBoundsException if offset or length or both would
 cause access outside array bounds.
- CRRP3: The method shall throw ME_PROFILE_NOT_AVAILABLE ToolkitException if Terminal Profile data are not available.

6.2.6.2.1.3 Context errors

No requirements.

6.2.6.2.2 Test suite files

Specific triggering:

UNFORMATTED_SMS_PP_UPD

No Additional requirements for the GSM personalization:

Test Script: API_2_MEP_CHEC_BSS_1.scr

Test Applet: API_2_MEP_CHEC_BSS_1.java

Load Script: API_2_MEP_CHEC_BSS_1.ldr (the applet is loaded without INI after the reset (RST))

Cleanup Script: API_2_MEP_CHEC_BSS_1.clr

Parameter File: API_2_MEP_CHEC_BSS_1.par

6.2.6.2.3 Test procedure

ld	Description	API Expectation	APDU
			Expectation
1	No Terminal Profile is registered Triggered by unformatted SMS Mask = 0xFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF	ME_PROFILE_NOT_AVAIL ABLE ToolkitException is thrown	
2	NULL as parameter to check	NullPointerException is	
	mask= NULL	thrown	
3	Offset > mask.length mask = 0xFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF	ArrayIndexOutOfBoundsExc eption is thrown	
4	Offset < 0 mask = 0xFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF	ArrayIndexOutOfBoundsExc eption is thrown	

5	Length > mask.length mask = 0xFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF	ArrayIndexOutOfBoundsExc eption is thrown
6	Offset + length > mask.length Mask = 0xFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF	ArrayIndexOutOfBoundsExc eption is thrown
7	<pre>length = 0 mask = 0xFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF</pre>	true is returned
8	Check all the Terminal Profile mask = 0xFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF	false is returned by the method because facility 15 is not supported
9	Check a part of the Terminal Profile mask = 0xFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF	true is returned by the method: the 16 first facilities except facility 15 have been successfully checked
10	Check a part of the Terminal Profile mask = 0x0080 Offset = 0 Length = 2	false is returned by the method only facility 15 is checked and not supported.

6.2.6.2.4 Test Coverage

CRR number	Test case number
N1	8, 9, 10
N2	7
P1	2
P2	3, 4, 5, 6
P3	1

6.2.6.3 Method check (short index)

Test Area Reference: API_2_MEP_CHECS

6.2.6.3.1 Conformance requirement:

The method with following header shall compliant to its definition in the API.

6.2.6.3.1.1 Normal execution

• CRRN1: The method checks a facility in the handset profile: returns true if the facility is supported, false if facility is not supported, or if facility-index outside MEProfile data.

6.2.6.3.1.2 Parameters error

No requirements.

6.2.6.3.1.3 Context errors

 CRRC1: The method shall throw ME_PROFILE_NOT_AVAILABLE ToolkitException if Terminal Profile data are not available

6.2.6.3.2 Test suite files

Specific triggering:

EVENT_STATUS_COMMAND

No Additional requirements for the GSM personalization:

Test Script: API_2_MEP_CHECS_1.scr

Test Applet: API_2_MEP_CHECS_1.java

Load Script: API_2_MEP_CHECS_1.ldr

Cleanup Script: API_2_MEP_CHECS_1.clr

Parameter File: API_2_MEP_CHECS_1.par

6.2.6.3.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	No Terminal Profile is registered	ME_PROFILE_NOT_AVAILABLE	
	Triggered by status command	ToolkitException is thrown	
	index = 1	·	
2	Terminal Profile, Facility is supported	true is returned by the method	
	index = 0	-	
3	Terminal Profile, Facility is not supported	false is returned by the method	
	index = 15		
4	Facility index is outside MEProfile data	false is returned by the method	
	index = 0x0099		

6.2.6.3.4 Test Coverage

CRR number	Test case number
N1	2, 3, 4
C1	1

6.2.6.4 Method getValue (short indexMSB, short indexLSB)

Test Area Reference: API_2_MEP_GVALSS

6.2.6.4.1 Conformance requirement:

The method with following header shall compliant to its definition in the API.

6.2.6.4.1.1 Normal execution

• CRRN1: The method returns the binary value of a parameter, delimited by two indexes, from the handset profile.

6.2.6.4.1.2 Parameters error

• CRRP1: The method shall throw BAD_INPUT_PARAMETER ToolkitException if (indexMSB indexLSB) or (indexMSB < 0) or (indexLSB < 0).

6.2.6.4.1.3 Context errors

• CRRC1: The method shall throw ME_PROFILE_NOT_AVAILABLE ToolkitException if Terminal Profile data are not available.

6.2.6.4.2 Test suite files

Specific triggering:

EVENT_STATUS_COMMAND

No Additional requirements for the GSM personalization:

Test Script: API_2_MEP_GVALSS_1.scr

Test Applet: API_2_MEP_GVALSS_1.java

Load Script: API_2_MEP_GVALSS_1.ldr

Cleanup Script: API_2_MEP_GVALSS_1.clr

Parameter File: API_2_MEP_GVALSS_1.par

6.2.6.4.3 Test procedure

TP = FF 01 D2 F0 00 00 00 00 00 00 00 00 00 8D FF

ld	Description	API Expectation	APDU Expectation
1	No Terminal Profile is registered	ME_PROFILE_NOT_AVAILABLE	-
	Triggered by status command	ToolkitException is thrown	
	indexMSB = 15, $indexLSB = 0$	·	
2	Retrieve number of character down ME display	13 is returned by the method	
	in Terminal Profile which is 13		
	indexMSB = 108, indexLSB = 104		
3	Retrieve byte 3 and byte 4 from terminal	0xF0D2 is returned by the method	
	profile.		
	Byte 3 = $0xD2$, Byte 4 = $0xF0$		
	indexMSB = 31, indexLSB = 16		
4	indexMSB is negative	BAD_INPUT_PARAMETER	
	indexMSB = 0xFFFF, indexLSB = 0xFFFD	ToolkitException is thrown	
5	indexLSB is negative	BAD_INPUT_PARAMETER	
	indexMSB = 0x0002, $indexLSB = 0xFFFD$	ToolkitException is thrown	
6	indexMSB < indexLSB	BAD_INPUT_PARAMETER	
	indexMSB = 0x0002, indexLSB = 0x0003	ToolkitException is thrown	
7	indexMSB > indexLSB + 16	BAD_INPUT_PARAMETER	
	indexMSB = 0x0021, $indexLSB = 0x0010$	ToolkitException is thrown	

6.2.6.4.4 Test Coverage

CRR number	Test case number
N1	2,3
P1	4, 5, 6, 7
C1	1

6.2.6.5 Method copy (short startOffset, byte[] dstBuffer, short dstOffset, short dstLength)

Test Area Reference: API_2_MEP_COPYS_BSS

6.2.6.5.1 Conformance requirement:

The method with following header shall compliant to its definition in the API.

 $\begin{array}{ll} {\tt public \ static \ short \ copy(short \ startOffset, \ byte[] \ dstBuffer, \ short \ dstOffset, \ short \ dstLength)} \\ {\tt throws \ \ } & \underline{{\tt ToolkitException}} \end{array}$

6.2.6.5.1.1 Normal execution

- CRRN1: The method copies a part of the handset profile in a buffer.
- CRRN2: The method returns dstOffset + dstLength.

6.2.6.5.1.2 Parameters error

- CRRP1: if dstBuffer is null NullPointerException is thrown.
- CRRP2: If dstOffset or dstLength parameter is negative an ArrayIndexOutOfBoundsException exception is thrown and no copy is performed
- CRRP3: If dstOffset+dstLength is greater than dstBuffer.length, the length of the dstBuffer array an ArrayIndexOutOfBoundsException exception is thrown and no copy is performed

6.2.6.5.1.3 Context errors

• CRRC1: The method shall throw ME_PROFILE_NOT_AVAILABLE ToolkitException if Terminal Profile data are not available.

6.2.6.5.2 Test suite files

Specific triggering:

EVENT_STATUS_COMMAND

No Additional requirements for the GSM personalization:

Test Script: API_2_MEP_COPYS_BSS_1.scr

Test Applet: API_2_MEP_COPYS_BSS_1.java

Load Script: API_2_MEP_COPYS_BSS_1.ldr

Cleanup Script: API_2_MEP_COPYS_BSS_1.clr

Parameter File: API_2_MEP_COPYS_BSS_1.par

6.2.6.5.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	No Terminal Profile is registered	ME_PROFILE_NOT_AVAILABLE	
	Triggered by status command	ToolkitException is thrown	
	startOffset = 0	·	
	dstBuffer.length = 6		
	dstOffset = 0		
	dstLength = 6		
2	dstBuffer is null	NullPointerException is thrown	
3	dstOffset ≥ dstBuffer.length	ArrayIndexOutOfBoundsException	
	startOffset = 0	is thrown	
	dstBuffer.length = 5		
	dstOffset = 5		
	dstLength = 1	1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2	
4	dstOffset < 0	ArrayIndexOutOfBoundsException	
	startOffset = 0	is thrown	
	dstBuffer.length = 5		
	dstOffset = -1		
5	dstLength = 1 dstLength < 0	ArrayIndexOutOfBoundsException	
3	startOffset = 0	is thrown	
	dstBuffer.length = 5	is tillowii	
	dstOffset = 1		
	dstLength = -1		
6	dstLength >dstBuffer.length	ArrayIndexOutOfBoundsException	
	startOffset = 0	is thrown	
	dstBuffer.length = 5		
	dstOffset = 0		
	dstLength = 6		
7	dstOffset + dstLength >dstBuffer.length	ArrayIndexOutOfBoundsException	
	startOffset = 0	is thrown	
	dstBuffer.length = 5		
	dstOffset = 3		
L	dstLength = 3		

ld	Description	API Expectation	APDU Expectation
8	Successful call extreme values	Result of copy() is 6	
	startOffset = 0		
	dstBuffer.length = 6		
	dstOffset = 0		
	dstLength = 6		
9	Successful call any values	Result of copy() is 7	
	startOffset = 1		
	dstBuffer.length = 20		
	dstOffset = 3		
	dstLength = 4		
10	Successful call, copy with length =0	Result of copy() is 20	
	startOffset = 0		
	dstBuffer.length = 20		
	dstOffset = 20		
	dstLength = 0		
11	Value outside MEProfile data available	Result of copy() is 6	
	startOffset = 13		
	dstBuffer.length = 6		
	dstOffset = 0		
	dstLength = 6		

6.2.6.5.4 Test Coverage

CRR number	Test case number
N1	8, 9, 10, 11
N3	8, 9, 10, 11
P1	2
P2	4, 5
P3	3, 6, 7
C1	1

6.2.7 Class ProactiveHandler

6.2.7.1 Method getTheHandler

Test Area Reference: API_2_PAH_GTHD

6.2.7.1.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

 $\label{eq:public_public} \mbox{public static ProactiveHandler getTheHandler()} \\ \mbox{throws ToolkitException}$

6.2.7.1.1.1 Normal execution

- CRRN1: The method shall return the single system instance of the ProactiveHandler class.
- CRRN2: The EnvelopeHandler is a Temporary JCRE Entry Point Object

6.2.7.1.1.2 Parameter errors

No requirements.

6.2.7.1.1.3 Context errors

• CRRC1: The method shall throw ToolkitException.HANDLER_NOT_AVAILABLE if the handler is busy.

6.2.7.1.2 Test Suite files

Test Script: API_2_PAH_GTHD_1.scr

Test Applet: API_2_PAH_GTHD_1.java

Load Script: API_2_PAH_GTHD_1.ldr

Cleanup Script: API_2_PAH_GTHD_1.clr

Parameter File: API_2_PAH_GTHD_1.par

6.2.7.1.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	getTheHandler() twice	The returned objects shall be the	
		same	
2	getTheHandler()	The reference shall be a	
		ProactiveHandler	
3	getTheHandler()	The reference shall not be null	

6.2.7.1.4 Test Coverage

CRR number Test case number	
N1	1, 2, 3
N2	To be checked in Framework tests and insert here cross reference
C1	To be checked in Framework tests and insert here cross reference

6.2.7.2 Method init

Test Area Reference: API_2_PAH_INITBBB

6.2.7.2.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

6.2.7.2.1.1 Normal execution

- CRRN1: The init() method initialises the next Proactive command in the ProactiveHandler, with Command details and Device Identities TLV. The source device is always the SIM Card (81h). The Comprehension Required flags are set.
- CRRN2: The Command number may take any value between 01h and FEh.
- CRRN3: The init() method clears the ProactiveHandler before initializing it.
- CRRN4: No TLV is selected after a call to the method.
- CRRN5: The handler is not sent to the mobile by the init() method.

6.2.7.2.1.2 Parameter errors

No requirements.

6.2.7.2.1.3 Context errors

No requirements.

6.2.7.2.2 Test Suite files

Test Script: API_2_PAH_INITBBB_1.scr

Test Applet: API_2_PAH_INITBBB_1.java

Load Script: API_2_PAH_INITBBB_1.ldr

Cleanup Script: API_2_PAH_INITBBB_1.clr

Parameter File: API_2_PAH_INITBBB_1.par

6.2.7.2.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Call the init() method type = 01h qualifier = 02h dstDevice = 03h Copy ProactiveHandler in a byte array (source)	source and reference are identical	
2	Verify the command number value	01h-FEh	
3	Call the init() method type = FFh qualifier = FEh destination = FDh Copy ProactiveHandler in a byte array (source)	source and reference are identical	
4	Select the 1st TLV in the handler Call the init() method with any value		
	Call the getValueLength() method	UNAVAILABLE_ELEMENT ToolkitException is thrown by getValueLength()	

6.2.7.2.4 Test Coverage

CRR number	Test case number	
N1	1, 3	
N2	2	
N3	3	
N4	4	
N5	1, 3	

6.2.7.3 Method initDisplayText

Test Area Reference: API_2_PAH_INDTBB_BSS

6.2.7.3.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

java.lang.ArrayIndexOutOfBoundsException,
ToolkitException

6.2.7.3.1.1 Normal execution

- CRRN1: The method shall build a DISPLAY TEXT proactive command in the ProactiveHandler, using qualifier, dcs and buffer parameters. Comprehension required flags are set.
- CRRN2: A call to this method clears the handler then initialises it.
- CRRN3: No TLV is selected after a call to the method.
- CRRN4: The DISPLAY TEXT command is not sent by the method.
- CRRN5: The Command Number may take any value between 01h and FEh.
- CRRN6: If length is equal to zero, then the Text String TLV inserted in the command is a null text string TLV as defined in TS 11.14 [4].

6.2.7.3.1.2 Parameter errors

- CRRP1: The method shall throw NullPointerException if buffer is null.
- CRRP2: If offset or length or both would cause access outside array bounds, an ArrayIndexOutOfBoundsException shall be thrown.

6.2.7.3.1.3 Context errors

• CRRC1: A ToolkitException.HANDLER_OVERFLOW shall be thrown if the ProactiveHandler is too small to put the requested data.

6.2.7.3.2 Test Suite files

Test Script: API_2_PAH_INDTBB_BSS_1.scr

Test Applet: API_2_PAH_INDTBB_BSS_1.java

Load Script: API_2_PAH_INDTBB_BSS_1.ldr

Cleanup Script: API_2_PAH_INDTBB_BSS_1.clr

Parameter File: API_2_PAH_INDTBB_BSS_1.par

6.2.7.3.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	NULL as parameter to buffer	NullPointerException is thrown	-
	buffer = NULL	·	
2	offset > buffer.length	ArrayIndexOutOfBoundsExceptio	
	buffer = "Text"	n is thrown	
	offset = 5		
	length = 0		
3	offset < 0	ArrayIndexOutOfBoundsExceptio	
	buffer = "Text"	n is thrown	
	offset = -1		
4	length > buffer.length	ArrayIndexOutOfBoundsExceptio	
	buffer = "Text"	n is thrown	
	offset = 0		
	length = 5		
5	offset + length > buffer.length	ArrayIndexOutOfBoundsExceptio	
	buffer = "Text"	n is thrown	
	offset = 3		
	length = 2		
6	length < 0	ArrayIndexOutOfBoundsExceptio	
	buffer = "Text"	n is thrown	
	offset = 3		
	length = -1		

ld	Description	API Expectation	APDU Expectation
7	Successful call, buffer is the whole buffer	No exception is thrown	2 5 Exposition
'	qualifier = 0	The exception is thown	
	dcs = 4		
	buffer = "TextA"		
	offset = 0		
	length = 5 Verify the command number value	Command number between 01h	
	verny the command number value	and FEh	
8	Send the command	and i Eii	DISPLAY TEXT Proactive
	Constitution of the consti		command
			ooniinana
			qualifier = 00h
			dcs = 4
9	Successful call, buffer is part of a buffer with		Text = "TextA" DISPLAY TEXT Proactive
9	the end part		command
	Send the command		Command
	qualifier = 0		qualifier = 00h
	dcs = 4		dcs = 4
	buffer = "12TextB"		Text = "TextB"
	offset = 2 length = 5		
10	Successful call, buffer is part of a buffer with		DISPLAY TEXT Proactive
	the first part		command
	Send the command		
	qualifier = 0		qualifier = 00h
	dcs = 4 buffer = "TextC12"		dcs = 4 Text = "TextC"
	offset = 0		
	length = 5		
11	Successful call, buffer is part of a buffer		DISPLAY TEXT Proactive
	Send the command		command
	qualifier = 0 dcs = 4		qualifier = 00h
	buffer = "12TextD34"		dcs = 4
	offset = 2		Text = "TextD"
L	length = 5		
12	Successful call, qualifier = 81h		DISPLAY TEXT Proactive
	Send the command qualifier = 81h		command
	dcs = 4		qualifier = 81h
	<pre>buffer = "TextE"</pre>		dcs = 4
	offset = 0		Text = "TextE"
13	length = 5 Successful call, DCS=0 (7 bits)		DISPLAY TEXT Proactive
13	Send the command		command
	qualifier = 0		Communa
	dcs = 0		qualifier = 00h
	buffer = "TextF"		dcs = 0
	offset = 0 length = 5		Text = "TextF"
14	Successful call, DCS=8 (UCS2)		DISPLAY TEXT Proactive
' *	Send the command		command
	qualifier = 0		
	dcs = 8		qualifier = 00h
	<pre>buffer = "TextG" offset = 0</pre>		dcs = 8 Text = "TextG"
	length = 5		TCAC - TEACG
15	Call the initDisplayText() method with any		DISPLAY TEXT Proactive
	value		command
	Then build and send a DISPLAY TEXT		1.61
	command		qualifier = 00h dcs = 4
	qualifier = 0 dcs = 4		Text = "TextHTextH"
	buffer = "TextHTextH"		
	offset = 0		
	length = 10		
16	Successful call, text length is zero		DISPLAY TEXT Proactive
	Send the command		command
	<pre>qualifier = 0 dcs = 4</pre>		qualifier = 00h
	buffer = "TextHTextH"		Text String TLV = 8D 00
	offset = 0		

ld	Description	API Expectation	APDU Expectation
	length = 0		
17	Select a TLV in the ProactiveHandler Call the initDisplayText() method Call the getValueLength() method	UNAVAILABLE_ELEMENT ToolkitException is thrown by getValueLength()	
18	Successful call, buffer length = 7Eh qualifier = 0		DISPLAY TEXT Proactive command
	<pre>dcs = 4 buffer = "UUU" offset = 0 length = 7Eh</pre>		Text String TLV = 8D 7F 04 55 55
19	Successful call, buffer length = 7Fh qualifier = 0		DISPLAY TEXT Proactive command
	<pre>dcs = 4 buffer = "UUU" offset = 0 length = 7Fh</pre>		Text String TLV = 8D 81 80 04 55 55
20	Successful call, buffer length = 240 Qualifier = 0		DISPLAY TEXT Proactive command
	<pre>dcs = 4 buffer = "UUU" offset = 0 length = 240</pre>		Text String TLV = 8D 81 F1 04 55 55
21	Call the initDisplayText() method with a too long buffer qualifier = 0 dcs = 4 buffer = "XXXX" offset = 0 length = 241	HANDLER_OVERFLOW ToolkitException is thrown	
22	Call the initDisplayText() without sending the command		No proactive command shall be sent expected status is '9000'

6.2.7.3.4 Test Coverage

CRR number	Test case number
N1	8, 9, 10, 11, 12, 13, 14, 15, 16, 18, 19, 20
N2	15
N3	17
N4	22
N5	7
N6	16
P1	1
P2	2, 3, 4, 5, 6
C1	21

6.2.7.4 Method initGetInkey

Test Area Reference: API_2_PAH_INGKBB_BSS

6.2.7.4.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

6.2.7.4.1.1 Normal execution

- CRRN1: The method shall build a GET INKEY proactive command in the ProactiveHandler, using qualifier, dcs and buffer parameters. Comprehension Required flags are set.
- CRRN2: A call to this method clears the handler then initialises it.
- CRRN3: No TLV is selected after a call to the method.
- CRRN4: The GET INKEY command is not sent by the method.
- CRRN5: The Command Number may take any value between 01h and FEh.
- CRRN6: If length is equal to zero, then the Text String TLV inserted in the command is a null text string TLV as defined in TS 11.14 [4].

6.2.7.4.1.2 Parameter errors

- CRRP1: The method shall throw NullPointerException if buffer is null.
- CRRP1: If offset or length or both would cause access outside array bounds, a ArrayIndexOutOfBoundsException shall be thrown.

6.2.7.4.1.3 Context errors

• CRRC1: A ToolkitException.HANDLER_OVERFLOW shall be thrown if the ProactiveHandler is to small to put the requested data.

6.2.7.4.2 Test Suite files

Test Script: API_2_PAH_INGKBB_BSS_1.scr

Test Applet: API_2_PAH_INGKBB_BSS_1.java

Load Script: API_2_PAH_INGKBB_BSS_1.ldr

Cleanup Script: API_2_PAH_INGKBB_BSS_1.clr

Parameter File: API_2_PAH_INGKBB_BSS_1.par

6.2.7.4.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	NULL as parameter to buffer	NullPointerException is thrown	
	buffer = NULL		
2	offset > buffer.length	ArrayIndexOutOfBoundsExceptio	
	buffer = "Text"	n is thrown	
	offset = 5		
3	offset < 0	ArrayIndexOutOfBoundsExceptio	
	buffer = "Text"	n is thrown	
	offset = -1		
4	length > buffer.length	ArrayIndexOutOfBoundsExceptio	
	buffer = "Text"	n is thrown	
	offset = 0		
	length = 5		
5	offset + length > buffer.length	ArrayIndexOutOfBoundsExceptio	
	buffer = "Text"	n is thrown	
	offset = 3		
	length = 2		
6	length < 0	ArrayIndexOutOfBoundsExceptio	
	buffer = "Text"	n is thrown	
	offset = 3		
	length = -1		
7	Successful call, buffer is the whole buffer	No exception is thrown	
	qualifier = 0		
	dcs = 4		
	buffer = "TextA"		

ld	Description	API Expectation	APDU Expectation
Iu	offset = 0	AFT EXPECTATION	AF DO EXPECTATION
	length = 5		
	Verify the command number value	Command number between 01h and FEh	
8	Send the command		GET INKEY Proactive
			command
			qualifier = 00h
			dcs = 4
9	Successful call, buffer is part of a buffer with		Text = "TextA" GET INKEY Proactive
9	the end part		command
	qualifier = 0		
	<pre>dcs = 4 buffer = "12TextB"</pre>		qualifier = 00h dcs = 4
	offset = 2		Text = "TextB"
	length = 5		
10	· •		GET INKEY Proactive
	the first part qualifier = 0		command
	dcs = 4		qualifier = 00h
	buffer = "TextC12"		dcs = 4
	offset = 0 length = 5		Text = "TextC"
11	Successful call, buffer is part of a buffer		GET INKEY Proactive
	Send the command		command
	qualifier = 0 dcs = 4		qualifier = 00h
	buffer = "12TextD34"		dcs = 4
	offset = 2		Text = "TextD"
12	length = 5 Successful call, qualifier = 81h		GET INKEY Proactive
12	qualifier = 81h		command
	dcs = 4		
	<pre>buffer = "TextE" offset = 0</pre>		qualifier = 81h
	length = 5		dcs = 4 Text = "TextE"
13	Successful call, DCS=0 (7 bits)		GET INKEY Proactive
	qualifier = 0 dcs = 0		command
	buffer = "TextF"		qualifier = 00h
	offset = 0 length = 5		dcs = 0
	Tengen - 5		Text = "TextF"
14	Successful call, DCS=8 (UCS2)		GET INKEY Proactive
	qualifier = 0		command
	<pre>dcs = 8 buffer = "TextG"</pre>		qualifier = 00h
	offset = 0		dcs = 8
	length = 5		Text = "TextG"
15	Call the initGetInkey() method with any value		GET INKEY Proactive
13	Then build and send a GET INKEY command		command
	qualifier = 0		
	<pre>dcs = 4 buffer = "TextHTextH"</pre>		qualifier = 00h
	offset = 0		Text = "TextHTextH"
	length = 10		OFT BUCEYO
16	Successful call, text length is zero Send the command		GET INKEY Proactive
	qualifier = 0		command
	dcs = 4		qualifier = 00h
	<pre>buffer = "TextHTextH" offset = 0</pre>		Text String TLV = 8D 00
L	length = 0		
17	Select a TLV in the ProactiveHandler	UNAVAILABLE_ELEMENT	
	Call the initGetInkey() method	ToolkitException is thrown by	
	Call the getValueLength() method	getValueLength()	
18	Successful call, buffer length = 7Eh		GET INKEY Proactive
			command

ld	Description	API Expectation	APDU Expectation
	qualifier = 0		
	dcs = 4		Text String TLV =
	buffer = "UUU"		8D 7F 04 55 55
	offset = 0		
	length = 7Eh		
19	Successful call, buffer length = 7Fh		GET INKEY Proactive
	1.5.		command
	qualifier = 0		
	dcs = 4 buffer = "UUU"		Text String TLV = 8D 81
	offset = 0		80 04 55 55
	length = 7Fh		
20	Successful call, buffer length = 240		GET INKEY Proactive
20	Ouccessial call, bullet lelight = 240		command
	Oualifier = 0		Command
	dcs = 4		Text String TLV =
	buffer = "UUU"		8D 81 F1 04 55 55
	offset = 0		
	length = 240		
21	Call the initGetInkey() method with a too long	HANDLER_OVERFLOW	
	buffer	ToolkitException is thrown	
	qualifier = 0	·	
	dcs = 4		
	buffer = "XXXX"		
	offset = 0		
	length = 241		
22	Call the initGetInkey() without sending the		No proactive command shall
	command		be sent expected status is
			'9000'

6.2.7.4.4 Test Coverage

CRR number	Test case number	
N1	8, 9, 10, 11, 12, 13, 14, 15, 16, 18, 19, 20	
N2	15	
N3	17	
N4	22	
N5	7	
N6	16	
P1	1	
P2	2, 3, 4, 5, 6	
C1	21	

6.2.7.5 Method initGetInput

Test Area Reference: API_2_PAH_INGPBB_BSSSS

6.2.7.5.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

6.2.7.5.1.1 Normal execution

• CRRN1: The method shall build a GET INPUT proactive command in the ProactiveHandler, using qualifier, dcs, buffer, minRespLength and maxRespLength parameters. Comprehension Required flags are set.

- CRRN2: A call to this method clears the handler then initialises it.
- CRRN3: No TLV is selected after a call to the method.
- CRRN4: The GET INPUT command is not sent by the method.
- CRRN5: The Command Number may take any value between 01h and FEh.
- CRRN6: If length is equal to zero, then the Text String TLV inserted in the command is a null text string TLV as defined in TS 11.14 [4].

6.2.7.5.1.2 Parameter errors

- CRRP1: The method shall throw NullPointerException if buffer is null.
- CRRP2: If offset or length or both would cause access outside array bounds, a ArrayIndexOutOfBoundsException shall be thrown.

6.2.7.5.1.3 Context errors

• CRRC1: A ToolkitException.HANDLER_OVERFLOW shall be thrown if the ProactiveHandler is to small to put the requested data.

6.2.7.5.2 Test Suite files

Test Script: API_2_PAH_INGPBB_BSSSS_1.scr

Test Applet: API_2_PAH_INGPBB_BSSSS_1.java

Load Script: API_2_PAH_INGPBB_BSSSS_1.ldr

Cleanup Script: API_2_PAH_INGPBB_BSSSS_1.clr

Parameter File: API_2_PAH_INGPBB_BSSSS_1.par

6.2.7.5.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	NULL as parameter to buffer	NullPointerException is thrown	
	buffer = NULL	·	
2	offset > buffer.length	ArrayIndexOutOfBoundsExceptio	
	buffer = "Text"	n is thrown	
	offset = 5		
3	offset < 0	ArrayIndexOutOfBoundsExceptio	
	buffer = "Text"	n is thrown	
	offset = -1		
4	length > buffer.length	ArrayIndexOutOfBoundsExceptio	
	buffer = "Text"	n is thrown	
	offset = 0		
	length = 5		
5	offset + length > buffer.length	ArrayIndexOutOfBoundsExceptio	
	buffer = "Text"	n is thrown	
	offset = 3		
6	length = 2	Arrayladay Out Of Bayada Evacatio	
О	length < 0	ArrayIndexOutOfBoundsExceptio	
	offset = 3	n is thrown	
	length = -1		
7	Successful call, buffer is the whole buffer	No exception is thrown	
-			
	qualifier = 0		
	dcs = 4		
	buffer = "TextA"		
	offset = 0		
	length = 5		
	minRespLength = 00h		
	maxRespLength = FFh		
	Verify the command number value	Command number between 01h	

ld	Description	API Expectation	APDU Expectation
	·	and FEh	-
8	Send the command		GET INPUT Proactive
			command
			qualifier = 00h
			dcs = 4
			Text = "TextA"
			Min Length = 00h Max Length = FFh
9	Successful call, buffer is part of a buffer with		GET INPUT Proactive
	the end part		command
	Send the command		qualifier = 00h
	qualifier = 0 dcs = 4		dcs = 4
	buffer = "12TextB"		Text = "TextB"
	offset = 2		Min Length = 10h
	<pre>length = 5 minRespLength = 10h</pre>		Max Length = FFh
	maxRespLength = FFh		
10	Successful call, buffer is part of a buffer with		GET INPUT Proactive
	the first part		command
	Send the command		qualifier = 00h
	qualifier = 0 dcs = 4		dcs = 4
	buffer = "TextC12"		Text = "TextC"
	offset = 0		Min Length = FFh
	<pre>length = 5 minRespLength = FFh</pre>		Max Length = FFh
	maxRespLength = FFh		
11	Successful call, buffer is part of a buffer		GET INPUT Proactive
	Send the command		command
	qualifier = 0 dcs = 4		qualifier = 00h
	buffer = "12TextD34"		dcs = 4
	offset = 2		Text = "TextD"
	length = 5		Min Length = 00h
	minRespLength = 00h maxRespLength = 00h		Max Length = 00h
12	Successful call, qualifier = 81h		GET INPUT Proactive
	qualifier = 81h		command
	<pre>dcs = 4 buffer = "TextE"</pre>		mualifier - 01h
	offset = 0		qualifier = 81h dcs = 4
	length = 5		Text = "TextE"
	minRespLength = 00h maxRespLength = 10h		Min Length = 00h
	Ton		Max Length = 10h
13	Successful call, DCS=0 (7 bits)		GET INPUT Proactive
	qualifier = 0		command
	<pre>dcs = 0 buffer = "TextF"</pre>		1.5.
	offset = 0		qualifier = 00h dcs = 0
	length = 5		Text = "TextF"
	minRespLength = 10h		Min Length = 10h
	maxRespLength = 10h		Max Length = 10h
14	Successful call, DCS=8 (UCS2)		GET INPUT Proactive
'-	qualifier = 0		command
	dcs = 8		
	<pre>buffer = "TextG" offset = 0</pre>		qualifier = 00h
	length = 5		dcs = 8 Text = "TextG"
	minRespLength = 00h		Min Length = 00h
	maxRespLength = FFh		Max Length = FFh
45	Call the init Cathemat () and the desired		OFT INDUT Description
15	Call the initGetInput() method with any value Then build and send a GET INPUT command		GET INPUT Proactive
	qualifier = 0		command
	dcs = 4		qualifier = 00h
	buffer = "TextHTextH"		dcs = 4
	offset = 0		Text = "TextHTextH"
	<pre>length = 10 minRespLength = 00h</pre>		Min Length = 00h Max Length = 10h
		1	1 2011-3011 1011

ld	Description	API Expectation	APDU Expectation
	maxRespLength = 10h		
16	Successful call, text length is zero Send the command qualifier = 0		GET INPUT Proactive command
	dds = 4 buffer = "TextHTextH" offset = 0		qualifier = 00h Text String TLV = 8D 00 Min Length = 00h
	<pre>length = 0 minRespLength = 00h maxRespLength = 10h</pre>		Max Length = 10h
17	Select a TLV in the ProactiveHandler Call the initGetInput() method	UNAVAILABLE_ELEMENT ToolkitException is thrown by	
	Call the getValueLength() method	getValueLength()	
18	Successful call, buffer length = 7Eh		GET INPUT Proactive command
	qualifier = 0 dcs = 4		Tout Chrina ELV
	buffer = "UUU"		Text String TLV = 8D 7F 04 55 55
	offset = 0		Min Length = 00h
	length = 7Eh		Max Length = 10h
	minRespLength = 00h		
	maxRespLength = 10h		
19	Successful call, buffer length = 7Fh		GET INPUT Proactive
			command
	qualifier = 0		
	dcs = 4 buffer = "UUU"		Text String TLV = 8D 81
	offset = 0		80 04 55 55
	length = 7Fh		Min Length = 00h Max Length = 10h
	minRespLength = 00h		Max Beligeli = 1011
	maxRespLength = 10h		
20	Successful call, buffer length = 236		GET INPUT Proactive command
	Qualifier = 0 dcs = 4		
	acs = 4 buffer = "UUU"		Text String TLV =
	offset = 0		8D 81 ED 04 55 55
	length = 236		
	minRespLength = 00h		
L	maxRespLength = 10h		
21	Call the initGetInput() method with a too long	HANDLER_OVERFLOW	
	buffer	ToolkitException is thrown	
	qualifier = 0		
	dcs = 4		
	buffer = "XXXX"		
	offset = 0		
	<pre>length = 237 minRespLength = 00h</pre>		
	maxRespLength = 10h		
22	Call the initGetInput() without sending the		No proactive command shall
	command		be sent expected status is '9000'

6.2.7.5.4 Test Coverage

CRR number	Test case number	
N1	8, 9, 10, 11, 12, 13, 14, 15, 16, 18, 19, 20	
N2	15	
N3	17	
N4	22	
N5	7	
N6	16	
P1	1	
P2	2, 3, 4, 5, 6	
C1	21	

6.2.7.6 Method send

Test Area Reference: API_2_PAH_SEND

6.2.7.6.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

public byte send()

6.2.7.6.1.1 Normal execution

- CRRN1: The send() method send the current proactive command to the mobile.
- CRRN2: The returned byte is equal to general result of the command (first byte of Result TLV in Terminal Response).
- CRRN3: The handler remains unchanged after a call to send() method until the use of initXX() or appendTLV().
- CRRN4: There is no invocation of select() or deselect() method.
- CRRN5: A pending toolkit applet transaction at the method invocation is aborted.

6.2.7.6.1.2 Parameter errors

No requirements.

6.2.7.6.1.3 Context errors

- CRRC1: A ToolkitException.UNAVAILABLE_ELEMENT shall be thrown is the Result Simple TLV is missing in Terminal Response.
- CRRC2: A ToolkitException.OUT_OF_TLV_BOUNDARIES shall be thrown if the general result byte is missing in the Result Simple TLV in Terminal Response.
- CRRC3: A ToolkitException COMMAND_NOT_ALLOWED shall be thrown if the proactive command to be sent is not allowed by the SIM Toolkit Framework.
- CRRC4: A ToolkitException COMMAND_NOT_ALLOWED shall be thrown if one parameter of the proactive command to be sent is not allowed by the SIM Toolkit Framework.

6.2.7.6.2 Test Suite files

Test Script: API_2_PAH_SEND_1.scr

Test Applet: API_2_PAH_SEND_1.java

Load Script: API_2_PAH_SEND_1.ldr

Cleanup Script: API_2_PAH_SEND_1.clr

Parameter File: API_2_PAH_SEND_1.par

6.2.7.6.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Build and send a DISPLAY TEXT command		DISPLAY TEXT Proactive
	qualifier = 00h		command
	dcs = 04h		
	buffer = 'Text'		
2	Terminal Response with General Result = 00	Result of send() is 00h	
	Result TLV = 03 01 00 (command performed		
	successfully)		
3	Build and send a DISPLAY TEXT command		DISPLAY TEXT Proactive
	qualifier = 00h		command
	dcs = 04h		

ld	Description	API Expectation	APDU Expectation
	buffer = 'Text'		,
4	Terminal Response with General Result = 01, without Additional information on result	Result of send() is 01h	
	Result TLV = 03 01 01 (command performed with partial comprehension)		
5	Build and send a DISPLAY TEXT command qualifier = 00h dcs = 04h		DISPLAY TEXT Proactive command
	buffer = 'Text'		
6	Terminal Response with General Result = 01, with Additional information on result	Result of send() is 01h	
	Result TLV = 03 02 01 55 (command performed with partial comprehension)		
7	Build and send a DISPLAY TEXT command qualifier = 00h dcs = 04h		DISPLAY TEXT Proactive command
<u> </u>	buffer = 'Text'	D 1/ (10 : 001	
8	Terminal Response with General Result = 02 Result TLV = 03 04 02 65 43 21 (Missing	Result of send() is 02h	
	information)		
9	Build and send a 7Fh byte command (DISPLAY TEXT) qualifier = 00h		DISPLAY TEXT Proactive command
	dcs = 04h		BER-TLV = D0 7F
	buffer = "UUUUU" length = 73h		Text String TLV = 8D 74 04 55 55 55
10	Build and send a 80h byte command (DISPLAY TEXT)		DISPLAY TEXT Proactive command
	qualifier = 00h dcs = 04h buffer = "UUUUU"		BER-TLV = D0 81 80 Text String TLV = 8D 75
	length = 74h		04 55 55 55
11	Build and send a maximum length command		DISPLAY TEXT Proactive
	(length of the handler should be 253)		command
	DISPLAY TEXT: Qualifier = 0		BER-TLV = D0 81 FD Text String TLV = 8D 81
	dcs = 4 buffer = "UUU"		F1 04 55 55
	offset = 0		
	length = 240		
12	Verify that the Proactive Handler is not		
	modified after a send() Build a DISPLAY TEXT command		
	Copy ProactiveHandler to source byte array		
	Send command		
	Copy ProactiveHandler to destination byte array		
	Compare source and destination	Source and destination are identical	
13	Build and send a DISPLAY TEXT command Verify there is no invocation of select() or	THO I HOU	DISPLAY TEXT Proactive command
4.4	deselect() method. Build and send a DISPLAY TEXT command		DISDLAY TEVE Dragative
14	Duild and Send a DISPLAY TEXT COMMAND		DISPLAY TEXT Proactive command
	Terminal Response with 2 Result TLV	Result of send() is 02h	
	1st Result TLV = 03 02 02 12 2nd Result TLV = 03 03 03 03 34 56		
15	Build and send a DISPLAY TEXT command		DISPLAY TEXT Proactive command
	Terminal Response without Result Simple TLV	ToolkitException.UNAVAILABLE	
			i .

ld	Description	API Expectation	APDU Expectation
		ELEMENT is thrown by send()	
16	Build and send a DISPLAY TEXT command		DISPLAY TEXT Proactive command
	Terminal Response without general result byte in the Simple TLV Result TLV = 03 00	ToolkitException.OUT_OF_TLV_ BOUNDARIES is thrown by send()	

6.2.7.6.4 Test Coverage

CRR number	Test case number
N1	1, 3, 5, 7, 9, 10, 11, 12, 13, 14
N2	2, 4, 6, 8, 14
N3	12
N4	13
N5	To be checked in Framework tests and insert here cross reference
C1	15
C2	16
C3	checked in the Framework test : FWK_PCS_PCCO (test case 1)
C4	checked in the Framework test : FWK_PCS_PCCO (test cases 2 to
	3)

6.2.7.7 Method getLength

Test Area Reference API_2_PAH_GLEN

6.2.7.7.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

6.2.7.7.1.1 Normal execution

• CRRN1: returns the length in bytes of the TLV list.

6.2.7.7.1.2 Parameter errors

No requirements.

6.2.7.7.1.3 Context errors

• CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException.HANDLER_NOT_AVAILABLE.

6.2.7.7.2 Test Suite files

Test Script: API_2_PAH_GLEN_1.scr

Test Applet: API_2_PAH_GLEN_1.java

Load Script: API_2_PAH_GLEN_1.ldr

Cleanup Script: API_2_PAH_GLEN_1.clr

Parameter File: API_2_PAH_GLEN_1.par

6.2.7.7.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Clear the handler getLength()	Result of getLength() is 0	
2	Call the init() method getLength()	Result of getLength() is 9	
3	Call the initDisplayText() method, with buffer length = 240 getLength()	Result of getLength() is 253	
4	Build a 7Fh Proactive Handler getLength()	Result of getLength() is 7Fh	
5	Build a 80h Proactive Handler getLength()	Result of getLength() is 80h	

6.2.7.7.4 Test Coverage

CRR number	Test case number
N1	1, 2, 3, 4, 5
C1	Does not apply for Proactive Handler

6.2.7.8 Method copy

Test Area Reference API_2_PAH_COPY_BSS

6.2.7.8.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

6.2.7.8.1.1 Normal execution

- CRRN1: copies the simple TLV list contained in the handler to the destination byte array.
- CRRN2: returns dstOffset + dstLength.

6.2.7.8.1.2 Parameter errors

- CRRP1: if dstBuffer is null a NullPointerException is thrown.
- CRRP2: if dstOffset or dstLength or both would cause access outside array bounds, or if dstLength is negative, an ArrayIndexOutOfBoundsException is thrown.
- CRRP3: if dstLength is grater than the length of the simple TLV List, an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException. OUT_OF_TLV_BOUNDARIES.

6.2.7.8.1.3 Context errors

• CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.

6.2.7.8.2 Test Suite files

Test Script: API_2_PAH_COPY_BSS_1.scr

Test Applet: API_2_PAH_COPY_BSS_1.java

Load Script: API_2_PAH_COPY_BSS_1.ldr

Cleanup Script: API_2_PAH_COPY_BSS_1.clr

Parameter File: API_2_PAH_COPY_BSS_1.par

6.2.7.8.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	NULL as parameter to dstBuffer	NullPointerException is thrown	-
2	Call the init() method		
	DstOffset > dstBuffer.length	ArrayIndexOutOfBoundsExceptio	
	dstBuffer.length = 5	n is thrown	
	dstOffset = 6 dstLength = 0		
3	dstOffset < 0	ArrayIndexOutOfBoundsExceptio	
3	dstBuffer.length = 5	n is thrown	
	dstOffset = -1	II IS UIIOWII	
	dstLength = 1		
4	DstLength > dstBuffer.length	ArrayIndexOutOfBoundsExceptio	
	dstBuffer.length = 5	n is thrown	
	dstOffset = 0		
5	dstLength = 6 dstOffset + dstLength > dstBuffer.length	ArrayladayOutOfPayadaEyaaatia	
Э	dstBuffer.length = 5	ArrayIndexOutOfBoundsException is thrown	
	dstOffset = 3	II is tillowii	
	dstLength = 3		
6	dstLength < 0	ArrayIndexOutOfBoundsExceptio	
	dstBuffer.length = 5	n is thrown	
	dstOffset = 0		
	dstLength = -1	TIIII OF TIV	
7	dstLength > length of the simple TLV list dstBuffer.length = 10	ToolkitException.OUT_OF_TLV_ BOUNDARIES is thrown	
	dstOffset = 0	BOUNDARIES IS INIOWN	
	dstLength = 10		
8	Successful call, dstBuffer is the whole buffer	Result of copy() is 9	
	dstBuffer.length = 9	.,,,	
	dstOffset = 0		
	dstLength = 9	D 11 (O O)	
9	Compare the buffer	Result of arrayCompare() is 0	
10	Successful call, dstBuffer is part of a buffer dstBuffer.length = 15	Result of copy() is 12	
	dstOffset = 3		
	dstLength = 9		
11	Compare the whole buffer	Result of arrayCompare() is 0	
12	Successful call, dstBuffer is part of a buffer	Result of copy() is 9	
	dstBuffer.length = 15		
	dstOffset = 3		
	dstLength = 6		
13	Compare the whole buffer	Result of arrayCompare() is 0	

6.2.7.8.4 Test Coverage

CRR number	Test case number	
N1	9, 11, 13	
N2	8, 10, 12	
P1	1	
P2	P2 2, 3, 4, 5, 6	
P3	7	
C1	Does not apply for ProactiveHandler	

6.2.7.9 Method findTLV

Test Area Reference API_2_PAH_FINDBB

6.2.7.9.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

6.2.7.9.1.1 Normal execution

Looks for the indicated occurrence of a TLV element from the beginning of the TLV list (handler buffer):

- CRRN1: the method is successful if the required occurrence exists then the corresponding TLV becomes current.
- CRRN2: if the method is successful then it returns TLV_FOUND_CR_SET when Comprehension Required flag is set.
- CRRN3: if the method is successful then it returns TLV_FOUND_CR_NOT_SET when Comprehension Required flag is not set.
- CRRN4: if the required occurrence of the TLV element does not exist, the current TLV is no longer defined and TLV NOT FOUND is returned.
- CRRN5: The search method is comprehension required flag independent.

6.2.7.9.1.2 Parameter errors

• CRRP1: if an input parameter is not valid (e.g. occurrence = 0) an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException BAD_INPUT_PARAMETER.

6.2.7.9.1.3 Context errors

• CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.

6.2.7.9.2 Test Suite files

Test Script: API 2 PAH FINDBB 1.scr

Test Applet: API_2_PAH_FINDBB_1.java

Load Script: API_2_PAH_FINDBB_1.ldr

Cleanup Script: API_2_PAH_FINDBB_1.clr

Parameter File: API_2_PAH_FINDBB_1.par

6.2.7.9.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Initialise the handler		
	Invalid input parameter	ToolkitException.BAD_INPUT_PA	
	Occurrence = 0	RAMETER is thrown	
2	Call the init() method		
	Search 1st TLV	Result is TLV_FOUND_CR_SET	
	Tag = 01h		
	Occurrence = 1		
3	Call the getValueLength() method	Result is 03h	
4	Search 2nd TLV	Result is TLV_FOUND_CR_SET	
	Tag = 02h		
	Occurrence = 1		
5	Call the getValueLength() method	Result is 02h	
6	Select a TLV (tag 02h)		
	Search a wrong tag	Result is TLV_NOT_FOUND	
	Tag = 03h		
	Occurrence = 1		

7	Call the getValueLength() method	ToolkitException.UNAVAILABLE	
		ELEMENT is thrown.	
8	Search a tag with wrong occurrence	Result is TLV_NOT_FOUND	
	Tag = 01h		
	Occurrence = 2		
		T HAS A LINIAN (AH ABI E	
9	Call the getValueLength() method	ToolkitException.UNAVAILABLE_	
		ELEMENT is thrown.	
10	Append a TLV with tag=02h		
10			
	Search the TLV	Result is	
	Tag = 02h	TLV FOUND CR NOT SET	
	Occurrence = 2	1212.001.0201.2101.202.	
11	Append a TLV with tag=04h		
	Search the TLV	Result is	
	Tag = 04h	TLV FOUND CR NOT SET	
	Occurrence = 1	1EV_1 00ND_0N_NO1_0E1	
12	Search tag 81h	Result is TLV_FOUND_CR_SET	
12	_	INESUIL IS TEV_TOOND_CIN_SET	
	Tag = 81h		
	Occurrence = 1		
13	Search tag 84h	Result is	
	Tag = 84h	TLV FOUND CR NOT SET	
	Occurrence = 1	ILV_I OUND_ON_INOT_OLT	
L	Occurrence - 1		

6.2.7.9.4 Test Coverage

CRR number	Test case number
N1	3, 5
N2	2, 4
N3	10, 11
N4	6, 7,8, 9
N5	12, 13
P1	1
C1	Does not apply for Proactive Handler

6.2.7.10 Method getValueLength

Test Area Reference API_2_PAH_GVLE

6.2.7.10.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

6.2.7.10.1.1 Normal execution

• CRRN1: gets and returns the binary length of the value field for the last TLV element which has been found in the handler.

6.2.7.10.1.2 Parameter errors

No requirements.

6.2.7.10.1.3 Context errors

- CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.
- CRRC2: in case of unavailable TLV element an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException UNAVAILABLE_ELEMENT.

6.2.7.10.2 Test Suite files

Test Script: API_2_PAH_GVLE_1.scr

Test Applet: API_2_PAH_GVLE_1.java

Load Script: API_2_PAH_GVLE_1.ldr

Cleanup Script: API_2_PAH_GVLE_1.clr

Parameter File: API_2_PAH_GVLE_1.par

6.2.7.10.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Call the init() method		
	getValueLength()	ToolkitException.UNAVAILABLE_ ELEMENT is thrown	
2	Call the appendTLV() method		
	tag = 0D		
	valueOffset = 0		
	valueLength = 0		
	Search TLV 0Dh (Text String TLV)		
	getValueLength()	Result is 00h	
3	Call the initDisplayText() method		
	length = 1 (+ dcs byte)		
	Search TLV 0Dh (Text String TLV)		
	getValueLength()	Result is 02h	
4	Call the initDisplayText() method		
	length = 7Eh (+ dcs byte)		
	Search TLV 0Dh (Text String TLV)		
	getValueLength()	Result is 7Fh	
5	Call the initDisplayText() method		
	length = 7Fh (+ dcs byte)		
	Search TLV 0Dh (Text String TLV)		
	getValueLength()	Result is 80h	
6	Call the initDisplayText() method		
	length = F0h (maximum text length)		
	Search TLV 0Dh (Text String TLV)		
	getValueLength()	Result is F1h	

6.2.7.10.4 Test Coverage

CRR number	Test case number	
N1	2, 3, 4, 5, 6	
C1	Does not apply for Proactive Handler	
C2	1	

6.2.7.11 Method getValueByte

Test Area Reference API_2_PAH_GVBYS

6.2.7.11.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

6.2.7.11.1.1 Normal execution

• CRRN1: Gets a byte from the last TLV element which has been found in the handler and returns its value (1 byte).

6.2.7.11.1.2 Parameter errors

• CRRP1: if valueOffset is out of the current TLV an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException OUT_OF_TLV_BOUNDARIES.

6.2.7.11.1.3 Context errors

- CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.
- CRRC2: in case of unavailable TLV element an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException UNAVAILABLE_ELEMENT.

6.2.7.11.2 Test Suite files

Test Script: API_2_PAH_GVBYS_1.scr

Test Applet: API_2_PAH_GVBYS_1.java

Load Script: API_2_PAH_GVBYS_1.ldr

Cleanup Script: API_2_PAH_GVBYS_1.clr

Parameter File: API_2_PAH_GVBYS_1.par

6.2.7.11.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Call the init() method		
	type = FFh		
	qualifier = FEh		
	destination = FDh	T 11:25 (2 11514) (411 45) 5	
	getValueByte(0)	ToolkitException.UNAVAILABLE_	
	0 1 711/04/40 10 10 4 11 7110	ELEMENT is thrown	
2	Search TLV 01h (Command Details TLV)		
	getValueByte(3)	ToolkitException.OUT_OF_TLV_	
		BOUNDARIES is thrown	
3	Search TLV 01h (Command Details TLV)		
	getValueByte(2)	Result is FEh (qualifier)	
4	Search TLV 02h (Device Identities TLV)		
	getValueByte(0)	Result is 81h (Source)	
5	<pre>initDisplayText() buffer = 00 01 7D length = 7Eh Search TLV 0Dh (Text String TLV)</pre>		
	getValueByte(7E)	Result is 7Dh	
6	<pre>initDisplayText() buffer = 00 01 7D 7E length = 7Fh Search TLV 0Dh (Text String TLV)</pre>		
	getValueByte(7E)	Result is 7Dh	
7	getValueByte(7F)	Result is 7Eh	
8	<pre>initDisplayText() buffer = 00 01 EF length = F0h Search TLV 0Dh (Text String TLV)</pre>		
	getValueByte(F0)	Result is EFh	

6.2.7.11.4 Test Coverage

CRR number	Test case number	
N1	3, 4, 5, 6, 7, 8	
P1	2	
C1	Does not apply for Proactive Handler	
C2	1	

6.2.7.12 Method copyValue

Test Area Reference API 2 PAH CPYVS BSS

6.2.7.12.1 Conformance requirement

The method with following header shall be compliant with its definition in the API.

6.2.7.12.1.1 Normal execution

- CRRN1: copies a part of the last TLV element which has been found, into a destination. buffer.
- CRRN2: returns dstOffset + dstLength.

6.2.7.12.1.2 Parameter errors

- CRRP1: if dstBuffer is null NullPointerException is thrown.
- CRRP2: if dstOffset or dstLength or both would cause access outside array bounds, or if dstLength is negative ArrayIndexOutOfBoundsException is thrown.
- CRRP3: if valueOffset is negative or valueOffset + dstLength > current TLV length, an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException OUT_OF_TLV_BOUNDARIES.

6.2.7.12.1.3 Context errors

- CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.
- CRRC2: in case of unavailable TLV element an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException UNAVAILABLE_ELEMENT.

6.2.7.12.2 Test Suite files

Test Script: API_2_PAH_CPYVS_BSS_1.scr
Test Applet: API_2_PAH_CPYVS_BSS_1.java
Load Script: API_2_PAH_CPYVS_BSS_1.ldr
Cleanup Script: API_2_PAH_CPYVS_BSS_1.clr
Parameter File: API_2_PAH_CPYVS_BSS_1.par

6.2.7.12.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Initialise the handler	7.1.7 Expodution	Do Expostation
1 '	Select a TLV		
	copyValue() with a null dstBuffer	NullPointerException is thrown	
2	initDisplayText() with length = 15		
	Select Text String TLV		
	dstOffset > dstBuffer.length	ArrayIndexOutOfBoundsExceptio	
	<pre>dstBuffer.length = 5 dstOffset = 6</pre>	n is thrown	
	dstLength = 0		
3	dstOffset < 0	ArrayIndexOutOfBoundsExceptio	
	dstBuffer.length = 5	n is thrown	
	dstOffset = -1		
4	dstLength = 1 dstLength >dstBuffer.length	ArrayIndexOutOfBoundsExceptio	
_	dstBuffer.length = 5	n is thrown	
	dstOffset = 0	ii io unowii	
	dstLength = 6		
5	dstOffset + dstLength >dstBuffer.length	ArrayIndexOutOfBoundsExceptio	
	<pre>dstBuffer.length = 5 dstOffset = 3</pre>	n is thrown	
	dstLength = 3		
6	dstLength < 0	ArrayIndexOutOfBoundsExceptio	
	dstBuffer.length = 5	n is thrown	
	<pre>dstOffset = 0 dstLength = -1</pre>		
	Labendell1		
7	initDisplayText() with length = 5		
	Select Text String TLV		
	valueOffset > Text String Length	ToolkitException.OUT_OF_TLV_	
	<pre>valueOffset = 7 dstBuffer.length = 15</pre>	BOUNDARIES is thrown	
	dstBuller.length = 15 dstOffset = 0		
L	dstLength = 0		
8	[Select Text String TLV]	ToolkitException.OUT_OF_TLV_	
	valueOffset < 0	BOUNDARIES is thrown	
	<pre>valueOffset = -1 dstBuffer.length = 15</pre>		
	dstOffset = 0		
	dstLength = 1		
9	[Select Text String TLV]	ToolkitException.OUT_OF_TLV_	
	<pre>dstLength > Text String length valueOffset = 0</pre>	BOUNDARIES is thrown	
	dstBuffer.length = 15		
	dstOffset = 0		
40	dstLength = 7	Tabliation of OUT OF THE	
10	[Select Text String TLV] valueOffset + dstLength > Text String	ToolkitException.OUT_OF_TLV_ BOUNDARIES is thrown	
	length	DOUNDAKIES IS (IIIOWI)	
	valueOffset = 2		
	dstBuffer.length = 15		
	<pre>dstOffset = 0 dstLength = 5</pre>		
11	Initialise the handler		
	copyValue()	ToolkitException.UNAVAILABLE_	
		ELEMENT is thrown	
12	initDisplayText()		
	dcs = 4 buffer = 00 01 0F		
	Select Text String TLV		
	Successful call	Result of copyValue() is 17	
	valueOffset = 0		
	<pre>dstBuffer.length = 17 dstOffset = 0</pre>		
	dstLength = 17		
13	Compare buffer	Result is 00h	
	buffer = 04 00 01 0F		
<u></u>			
14	initialise dstBuffer		
\vdash	dstBuffer = 55 55 55 Successful call	Result of copyValue() is 15	
1	Successiui cali	Inesult of copyvalue() is 15	

ld	Description	API Expectation	APDU Expectation
	valueOffset = 2		
	dstBuffer.length = 20		
	dstOffset = 3		
	dstLength = 12		
15	Compare buffer	Result is 00h	
	buffer =		
	55 55 55 01 02		
	03 04 05 06 07		
	08 09 0A 0B 0C		
	55 55 55 55		

6.2.7.12.4 Test Coverage

CRR number	Test case number
N1	13, 15
N2	12, 14
P1	1
P2	2, 3, 4, 5, 6
P3	7, 8, 9, 10
C1	Does not apply for Proactive Handler
C2	11

6.2.7.13 Method compare Value

Test Area Reference API_2_PAH_CPRVS_BSS

6.2.7.13.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

6.2.7.13.1.1 Normal execution

Compares the last found TLV element with a buffer:

- CRRN1: returns 0 if identical.
- CRRN2: returns -1 if the first miscomparing byte in simple TLV List is less than that in compareBuffer.
- CRRN3: returns 1 if the first miscomparing byte in simple TLV List is greater than that in compareBuffer.

6.2.7.13.1.2 Parameter errors

- CRRP1: if compareBuffer is null NullPointerException shall be thrown.
- CRRP2: if compareOffset or compareLength or both would cause access outside array bounds, or if compareLength is negative ArrayIndexOutOfBoundsException shall be thrown.
- CRRP3: if valueOffset is negative or valueOffset + dstLength > current TLV length, an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException OUT_OF_TLV_BOUNDARIES.

6.2.7.13.1.3 Context errors

• CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.

• CRRC2: in case of unavailable TLV element an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException UNAVAILABLE_ELEMENT.

6.2.7.13.2 Test Suite files

Test Script: API_2_PAH_CPRVS_BSS_1.scr

Test Applet: API_2_PAH_CPRVS_BSS_1.java

Load Script: API_2_PAH_CPRVS_BSS_1.ldr

Cleanup Script: API_2_PAH_CPRVS_BSS_1.clr

Parameter File: API_2_PAH_CPRVS_BSS_1.par

6.2.7.13.3 Test procedure

la!	Description	ADI Evmostation	ADDII Everatetias
ld	Description	API Expectation	APDU Expectation
1	Initialise the handler Select a TLV		
		NullDeinterEvention in through	
	compareValue() with a null compareBuffer	NullPointerException is thrown	
2	initDisplayText() with length = 15		
_	Select Text String TLV		
	compareOffset > compareBuffer.length	ArrayIndexOutOfBoundsExceptio	
	compareBuffer.length = 5	n is thrown	
	compareOffset = 6		
	compareLength = 0		
3	compareOffset < 0	ArrayIndexOutOfBoundsExceptio	
	compareBuffer.length = 5	n is thrown	
	compareOffset = -1		
	compareLength = 1	A 1 1 0 10'D 1 5 1:	
4	compareLength >compareBuffer.length	ArrayIndexOutOfBoundsExceptio	
	<pre>compareBuffer.length = 5 compareOffset = 0</pre>	n is thrown	
	compareLength = 6		
5	compareOffset + compareLength	ArrayIndexOutOfBoundsExceptio	
Ü	>compareBuffer.length	n is thrown	
	compareBuffer.length = 5	ii io unown	
	compareOffset = 3		
	compareLength = 3		
6	compareLength < 0	ArrayIndexOutOfBoundsExceptio	
	compareBuffer.length = 5	n is thrown	
	compareOffset = 0		
	compareLength = -1		
7	initDisplayText() with length = 5		
	Select Text String TLV		
	valueOffset > Text String Length	ToolkitException.OUT_OF_TLV_	
	valueOffset = 7	BOUNDARIES is thrown	
	compareBuffer.length = 15		
	compareOffset = 0		
_	compareLength = 0	Tablist Cycontian OUT OF TIV	
8	[Select Text String TLV]	ToolkitException.OUT_OF_TLV_	
	valueOffset = -1	BOUNDARIES is thrown	
	compareBuffer.length = 15		
	compareOffset = 0		
	compareLength = 1		
9	[Select Text String TLV]	ToolkitException.OUT_OF_TLV_	
	compareLength > Text String length	BOUNDARIES is thrown	
	valueOffset = 0		
	compareOffget = 0		
	<pre>compareOffset = 0 compareLength = 7</pre>		
10	[Select Text String TLV]	ToolkitException.OUT_OF_TLV_	
10	valueOffset + compareLength > Text String	BOUNDARIES is thrown	
	length	POOMPANIES IS IIIIUWII	
	valueOffset = 2		
	compareBuffer.length = 15		
	compareOffset = 0		
	compareLength = 5		

ld	Description	API Expectation	APDU Expectation
	Boothphon	7.1 Expediation	A DO EXPONENTIA
11	Initialise the handler		
	compareValue()	ToolkitException.UNAVAILABLE_ ELEMENT is thrown	
12	initDisplayText()		
	dcs = 4 buffer = 00 01 0F		
	Select Text String TLV		
	Initialise compareBuffer		
	compareBuffer =		
	04 00 01 0F Compare buffers	Result is 00h	
	valueOffset = 0	Result is oon	
	compareOffset = 0		
	compareLength = 17		
13	Initialise compareBuffer		
	compareBuffer =		
	04 00 01 02 03 04 05 06 07 08		
	05 0A 0B 0C 0D		
	0E 10		
	Compare buffers with same parameters	Result is -1	
14	Initialise compareBuffer		
	compareBuffer = 03 00 01 0F		
	Compare buffers with same parameters	Result is +1	
15	Initialise compareBuffer		
	compareBuffer = 55		
	03 04 05 06 07		
	08 09 0A 0B 0C		
	55 55 55 55 55	Result is 00h	
	valueOffset = 2	Tresult is out	
	compareOffset = 3		
	compareLength = 12		
16	Initialise compareBuffer		
_	compareBuffer =		
	55 55 55 02 01 03 04 05 06 07		
	08 09 0A 0B 0C		
	55 55 55 55		
	Compare buffers with same parameters	Result is -1	
17	Initialise compareBuffer		
	compareBuffer =		
	55		
	08 09 0A 0A 0D		
	55 55 55 55		
10	Compare buffers with same parameters Initialise compareBuffer	Result is +1	
18	compareBuffer =		
	55 55 55 99 03		
	03 04 05 06 07		
	08 09 0A 0B 0C 55 55 55 55		
	Compare buffers with same parameters	Result is +1	
		1	

6.2.7.13.4 Test Coverage

CRR number	Test case number
N1	12, 15
N2	13, 16
N3	14, 17, 18
P1	1
P2	2, 3, 4, 5, 6
P3	7, 8, 9, 10
C1	Does not apply for Proactive Handler
C2	11

6.2.7.14 Method findAndCopyValue(byte tag, byte[] dstBuffer, short valueOffset)

Test Area Reference API_2_PAH_FACYB_BS

6.2.7.14.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

6.2.7.14.1.1 Normal execution

- CRRN1: looks for the first occurrence of a TLV element from the beginning of a TLV list and copy its value into a destination buffer.
- CRRN2: if no TLV element is found, the UNAVAILABLE_ELEMENT exception is thrown and the current TLV is no longer defined.
- CRRN3: if the method is successful then the corresponding TLV becomes current and dstOffset + length of the copied value is returned.
- CRRN4: The search method is comprehension required flag independent.

6.2.7.14.1.2 Parameter errors

- CRRP1: if dstBuffer is null NullPointerException shall be thrown.
- CRRP2: if dstOffset would cause access outside array bounds ArrayIndexOutOfBoundsException shall be thrown.

6.2.7.14.1.3 Context errors

• CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.

6.2.7.14.2 Test Suite files

Test Script: API_2_PAH_FACYB_BS_1.scr
Test Applet: API_2_PAH_FACYB_BS_1.java
Load Script: API_2_PAH_FACYB_BS_1.ldr
Cleanup Script: API_2_PAH_FACYB_BS_1.clr
Parameter File: API_2_PAH_FACYB_BS_1.par

6.2.7.14.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Initialise the handler		
-	FindAndCopyValue() with a null dstBuffer	NullPointerException is thrown	
2	InitDisplayText() with length = 15		
	dstOffset > dstBuffer.length	ArrayIndexOutOfBoundsExceptio	
	tag = 0Dh dstBuffer.length = 20 dstOffset = 21	n is thrown	
3	dstOffset < 0	ArrayIndexOutOfBoundsExceptio	
	dstBuffer.length = 20 dstOffset = -1	n is thrown	
4	<pre>length > dstBuffer.length dstBuffer.length = 15 dstOffset = 0</pre>	ArrayIndexOutOfBoundsException is thrown	
5	DstOffset + length >dstBuffer.length DstBuffer.length = 20 DstOffset = 5	ArrayIndexOutOfBoundsExceptio n is thrown	
6	initDisplayText()		
	Select a TLV (tag 02h)		
	findAndCopyValue() tag = 03h	ToolkitException.UNAVAILABLE_ ELEMENT is thrown	
	Call the getValueLength() method	ToolkitException.UNAVAILABLE_ ELEMENT is thrown.	
7	initDisplayText() dcs = 4 buffer = 00 01 0F		
	Successful call Tag = 0Dh DstBuffer.length = 17 DstOffset = 0	Result of findAndcopyValue() is 17	
8	Compare buffer buffer = 04 00 01 OF	Result is 00h	
9	initialise dstBuffer dstBuffer = 55 55 55		
	Successful call dstBuffer.length = 20 dstOffset = 2	Result of findAndcopyValue() is 19	
10	Compare buffer	Result is 00h	
	buffer = 55 55 04 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F 55		
11	initDisplayText() dcs = 4 buffer = 00 01 0F		
	append a 2nd Text String TLV		
	Successful call tag = 0Dh dstBuffer.length = 17 dstOffset = 0	Result of findAndcopyValue() is 17	
12	Compare buffer buffer = 04 00 01 0F	Result is 00h	
13	initDisplayText() dcs = 4 buffer = 00 01 0F		
	Successful call (with tag 8Dh) tag = 8Dh dstBuffer.length = 17 dstOffset = 0	Result of findAndcopyValue() is 17	
14	Compare buffer buffer = 04 00 01 0F	Result is 00h	

ld	Description	API Expectation	APDU Expectation
15	Append tag 0Fh		
	buffer = 00 01 0F		
	Successful call (with tag 8Fh)	Result of findAndcopyValue() is	
	tag = 8Fh	16	
	dstBuffer.length = 16		
	dstOffset = 0		
16	Compare buffer	Result is 00h	
	buffer = 00 01 0F		

6.2.7.14.4 Test Coverage

CRR number	Test case number
N1	8, 10, 12
N2	6
N3	7, 9, 11
N4	13, 14, 15, 16
P1	1
P2	2, 3, 4, 5
C1	Does not apply for Proactive Handler

6.2.7.15 Method findAndCopyValue(byte tag, byte occurence, short valueOffset, byte[] dstBuffer, short dstOffset, short dstLength)

Test Area Reference API_2_PAH_FACYBBS_BSS

6.2.7.15.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

6.2.7.15.1.1 Normal execution

- CRRN1: looks for the indicated occurrence of a TLV element from the beginning of a TLV list and copy its value into a destination buffer.
- CRRN2: if no TLV element is found, the UNAVAILABLE_ELEMENT exception is thrown and the current TLV is no longer defined.
- CRRN3: if the method is successful then the corresponding TLV becomes current and dstOffset + dstLength is returned.
- CRRN4: The search method is comprehension required flag independent.

6.2.7.15.1.2 Parameter errors

- CRRP1: if dstBuffer is null NullPointerException shall be thrown.
- CRRP2: if dstOffset or dstLength or both would cause access outside array bounds, or if dstLength is negative ArrayIndexOutOfBoundsException shall be thrown.
- CRRP3: if valueOffset is negative or valueOffset + dstLength > current TLV length, an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException OUT_OF_TLV_BOUNDARIES.

6.2.7.15.1.3 Context errors

• CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.

6.2.7.15.2 Test Suite files

Test Script: API_2_PAH_FACYBBS_BSS_1.scr

Test Applet: API_2_PAH_FACYBBS_BSS_1.java

Load Script: API_2_PAH_FACYBBS_BSS_1.ldr

Cleanup Script: API_2_PAH_FACYBBS_BSS_1.clr

Parameter File: API_2_PAH_FACYBBS_BSS_1.par

6.2.7.15.3 Test procedure

	5	ABI E	ADDILE
ld	Description	API Expectation	APDU Expectation
11	Initialise the handler		
	findAndCopyValue() with a null dstBuffer	NullPointerException is thrown	
	1.10 1.1 T. (0. 11 L. (1. 45		
2	initDisplayText() with length = 15		
	dstOffset > dstBuffer.length	ArrayIndexOutOfBoundsExceptio	
	tag = 0Dh, occurrence = 1	n is thrown	
	<pre>valueOffset = 0 dstBuffer.length = 5</pre>		
	dstBuller.length = 5 dstOffset = 6		
	dstLength = 0		
3	dstOffset < 0	ArrayIndexOutOfBoundsExceptio	
3	dstBuffer.length = 5	n is thrown	
	dstOffset = -1	II is unown	
	dstLength = 1		
4	dstLength >dstBuffer.length	ArrayIndexOutOfBoundsExceptio	
	dstBuffer.length = 5	n is thrown	
	dstOffset = 0		
	dstLength = 6		
5	dstOffset + dstLength >dstBuffer.length	ArrayIndexOutOfBoundsExceptio	
	dstBuffer.length = 5	n is thrown	
	dstOffset = 3		
	dstLength = 3		
6	dstLength < 0	ArrayIndexOutOfBoundsExceptio	
	dstBuffer.length = 5	n is thrown	
	<pre>dstOffset = 0 dstLength = -1</pre>		
	dischengen = -1		
7	initDisplayText() with length = 5		
	valueOffset > Text String Length	ToolkitException.OUT_OF_TLV_	
	tag = 0Dh, occurrence = 1	BOUNDARIES is thrown	
	valueOffset = 7		
	dstBuffer.length = 15		
	dstOffset = 0		
_	dstLength = 0 valueOffset < 0	ToolkitEvention OUT OF TIV	
8	valueOffset = -1	ToolkitException.OUT_OF_TLV_ BOUNDARIES is thrown	
	dstBuffer.length = 15	DOUNDARIES IS UIIOWII	
	dstOffset = 0		
	dstLength = 1		
9	dstLength > Text String length	ToolkitException.OUT_OF_TLV_	
	valueOffset = 0	BOUNDARIES is thrown	
	dstBuffer.length = 15		
	dstOffset = 0		
<u> </u>	dstLength = 7		
10	valueOffset + dstLength > Text String length	ToolkitException.OUT_OF_TLV_	
	valueOffset = 2	BOUNDARIES is thrown	
	dstBuffer.length = 15		
	dstOffset = 0		
	dstLength = 5		
<u> </u>			

ld	Description	API Expectation	APDU Expectation
11	InitDisplayText()	AFIEXPECIATION	APDO Expectation
11	Select a TLV (tag 02h)		
	findAndCopyValue()	ToolkitException.UNAVAILABLE_	
	tag = 0Dh	ELEMENT is thrown	
	occurrence = 2	ELEMENT IS UNOWN	
	Call the getValueLength() method	ToolkitException.UNAVAILABLE_	
		ELEMENT is thrown.	
12	initDisplayText()		
	dcs = 4		
	buffer = 00 01 0F	D - f f -	
	Successful call tag = 0Dh, occurrence = 1	Result of findAndCopyValue() is 17	
	valueOffset = 0	17	
	dstBuffer.length = 17		
	dstOffset = 0		
40	dstLength = 17	Decult is 00h	
13	Compare buffer buffer = 04 00 01 0F	Result is 00h	
	Buller - 04 00 01 0r		
14	initialise dstBuffer		
	dstBuffer = 55 55 55		
	Successful call	Result of findAndcopyValue() is	
	tag = 0Dh, occurrence = 1	15	
	<pre>valueOffset = 2 dstBuffer.length = 20</pre>		
	dstOffset = 3		
	dstLength = 12		
15	Compare buffer	Result is 00h	
	buffer = 55 55 55 01 02		
	03 04 05 06 07		
	08 09 0A 0B 0C		
	55 55 55 55		
16	Append a Text String TLV		
	tag = 0D buffer = 00 11 22 33 44 55 (no specific		
	DCS byte)		
	Successful call	Result of findAndCopyValue() is	
	tag = 0Dh, occurrence = 1	17	
	<pre>valueOffset = 0 dstBuffer.length = 17</pre>		
	dstBuller.length = 17 dstOffset = 0		
	dstLength = 17		
17	Compare buffer	Result is 00h	
	buffer = 04 00 01 0F		
18	Successful call	Result of findAndCopyValue() is 6	
10	tag = 0Dh, occurrence = 2	Todait of infuctioopy value() is 6	
	valueOffset = 0		
	dstBuffer.length = 6		
	dstOffset = 0 dstLength = 6		
19	Compare buffer	Result is 00h	
	buffer = 00 11 22 33 44 55		
20	initDisplayText()		
	dcs = 4 buffer = 00 01 0F		
	Successful call (with tag 8Dh)	Result of findAndcopyValue() is	
	tag = 8Dh	17	
	occurrence = 1		
	<pre>valueOffset = 0 dstBuffer.length = 17</pre>		
	dstOffset = 0		
	dstLength = 17		
21	Compare buffer	Result is 00h	
	buffer = 04 00 01 0F		
i	I .	i	

ld	Description	API Expectation	APDU Expectation
22	Append tag 0Fh		
	buffer = 00 01 0F		
	Successful call (with tag 8Fh)	Result of findAndcopyValue() is	
	tag = 8Fh	16	
	occurrence = 1		
	valueOffset = 0		
	dstBuffer.length = 16		
	dstOffset = 0		
	dstLength = 16		
23	Compare buffer	Result is 00h	
	buffer = 00 01 0F		

6.2.7.15.4 Test Coverage

CRR number	Test case number
N1	13, 15, 17, 19
N2	11
N3	12, 14, 16, 18
N4	20, 21, 22, 23
P1	1
P2	2, 3, 4, 5, 6
P3	7, 8, 9, 10
C1	Does not apply for ProactiveHandler

6.2.7.16 Method findAndCompareValue(byte tag, byte[] compareBuffer, short compareOffset)

Test Area Reference API_2_PAH_FACRB_BS

6.2.7.16.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

6.2.7.16.1.1 Normal execution

Looks for the first occurrence of a TLV element from beginning of a TLV list and compare its value with a buffer:

- CRRN1: if no TLV element is found, the UNAVAILABLE_ELEMENT exception is thrown and the current TLV is no longer defined.
- CRRN2: if the method is successful then the corresponding TLV becomes current.
- CRRN3: if identical returns 0.
- CRRN4: if the first miscomparing byte in simple TLV is less than that in compareBuffer returns -1.
- CRRN5: if the first miscomparing byte in simple TLV is greater than that in compareBuffer returns 1.
- CRRN6: The search method is comprehension required flag independent.

6.2.7.16.1.2 Parameter errors

• CRRP1: if compareBuffer is null NullPointerException shall be thrown.

• CRRP2: if compareOffset would cause access outside array bounds ArrayIndexOutOfBoundsException shall be thrown.

6.2.7.16.1.3 Context errors

• CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.

6.2.7.16.2 Test Suite files

Test Script: API_2_PAH_FACRB_BS_1.scr

Test Applet: API_2_PAH_FACRB_BS_1.java

Load Script: API_2_PAH_FACRB_BS_1.ldr

Cleanup Script: API_2_PAH_FACRB_BS_1.clr

Parameter File: API_2_PAH_FACRB_BS_1.par

6.2.7.16.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Initialise the handler		
	findAndCompareValue() with a null dstBuffer	NullPointerException is thrown	
2	initDisplayText() with length = 15		
	<pre>compareOffset > compareBuffer.length tag = 0Dh compareBuffer.length = 20 compareOffset = 21</pre>	ArrayIndexOutOfBoundsExceptio n is thrown	
3	<pre>compareOffset < 0 compareBuffer.length = 20 compareOffset = -1</pre>	ArrayIndexOutOfBoundsExceptio n is thrown	
4	<pre>length > compareBuffer.length compareBuffer.length = 15 compareOffset = 0</pre>	ArrayIndexOutOfBoundsExceptio n is thrown	
5	<pre>compareOffset + length > compareBuffer.length compareBuffer.length = 20 compareOffset = 5</pre>	ArrayIndexOutOfBoundsExceptio n is thrown	
6	InitDisplayText()		
	Select a TLV (tag 02h)		
	findAndCompareValue() tag = 03h	ToolkitException.UNAVAILABLE_ ELEMENT is thrown	
	Call the getValueLength() method	ToolkitException.UNAVAILABLE_ ELEMENT is thrown.	
7	initDisplayText() dcs = 4 buffer = 00 01 0F		
	Initialise compareBuffer compareBuffer = 04 00 01 0F		
	Compare buffers tag = 0Dh compareOffset = 0	Result is 00h	
8	Verify current TLV getValueLength()	Result is 17	
9	Initialise compareBuffer compareBuffer = 04 00 01 10		
	Compare buffers with same parameters	Result is -1	
10	Initialise compareBuffer compareBuffer = 03 00 01 0F		
	Compare buffers with same parameters	Result is +1	

ld	Description	API Expectation	APDU Expectation
11	Initialise compareBuffer		
1	compareBuffer =		
	55 55 04 00 01		
	02 03 04 05 06		
	07 08 09 0A 0B 0C 0D 0E 0F 55		
	Compare buffers	Result is 00h	+
	compare buffers compareOffset = 2	Nesult is UUII	
ĺ			
12	append a Text String TLV		
	tag = 0Dh		
	buffer = 00 11 22 33 44 55		
	Initialise compareBuffer		
1	compareBuffer =		
1	55 55 04 00 01 02 03 04 05 06		
1	07 08 09 0A 0B		
L	OC OD OE OF 55		
	Compare buffers	Result is 00h	
	compareOffset = 2		
<u> </u>			
13	Initialise compareBuffer		
	compareBuffer = 55 55 04 01 01		
	02 03 04 05 06		
	07 08 09 0A 0B		
	OC OD OE OF 55		
	Compare buffers	Result is -1	
	compareOffset = 2		
4.4	Initialiae commans Buff		
14	Initialise compareBuffer compareBuffer =		
	compareBuller		
	02 03 04 05 06		
	07 08 09 0A 0B		
-	OC OD OD 10 55	Population : 4	
	Compare buffers compareOffset = 2	Result is +1	
15	initDisplayText()		
-	dcs = 4		
	buffer = 00 01 0F		
	Initialise compareBuffer		1
	CompareBuffer = 04 00 01 0F	Popult in COL	_
	Successful call (with tag 8Dh)	Result is 00h	
	<pre>tag = 8Dh compareBuffer.length = 17</pre>		
	compareOffset = 0		
16	Append tag 0Fh		
<u></u>	buffer = 00 01 0F		
_	Initialise compareBuffer		
<u> </u>	compareBuffer = 00 01 0F	Popult is 00t	_
	Successful call (with tag 8Fh) tag = 8Fh	Result is 00h	
	<pre>tag = 8Fh compareBuffer.length = 16</pre>		l l
	compareOffset = 0		
17	Initialise compareBuffer		
	compareBuffer = 00 99 01 03 OF		
-	Successful call (with tag 8Fh)	Result is +1	
	tag = 8Fh		
1	<pre>compareBuffer.length = 16 compareOffset = 0</pre>		
	COMPATENTIBEL - 0		

6.2.7.16.4 Test Coverage

CRR number	Test case number
N1	6
N2	8
N3	7, 11, 12, 17
N4	9, 13

CRR number	Test case number
N5	10, 14
N6	15, 16
P1	1
P2	2, 3, 4, 5
C1	Does not apply for Proactive Handler

6.2.7.17 Method findAndCompareValue(byte tag, byte occurence, short valueOffset, byte[] compareBuffer, short compareOffset, short compareLength)

Test Area Reference API 2 PAH FACRBBS BSS

6.2.7.17.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

6.2.7.17.1.1 Normal execution

Looks for the indicated occurrence of a TLV element from the beginning of a TLV list and compare its value with a buffer:

- CRRN1: if no TLV element is found, the UNAVAILABLE_ELEMENT exception is thrown and the current TLV is no longer defined.
- CRRN2: if the method is successful then the corresponding TLV becomes current.
- CRRN3: if identical 0 is returned.
- CRRN4: if the first miscomparing byte in simple TLV is less than that in compareBuffer -1 is returned.
- CRRN5: if the first miscomparing byte in simple TLV is greater than that in compareBuffer 1 is returned.
- CRRN6: The search method is comprehension required flag independent.

6.2.7.17.1.2 Parameter errors

- CRRP1: if compareBuffer is null NullPointerException shall be thrown.
- CRRP2: if compareOffset or compareLength or both would cause access outside array bounds, or if compareLength is negative ArrayIndexOutOfBoundsException shall be thrown.
- CRRP3: if valueOffset is negative or valueOffset + dstLength > current TLV length, an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException OUT_OF_TLV_BOUNDARIES.
- CRRP4: if an input parameter is not valid (e.g. occurrence = 0) an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException BAD_INPUT_PARAMETER.

6.2.7.17.1.3 Context errors

 CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE. 6.2.7.17.2 Test Suite files

Test Script: API_2_PAH_FACRBBS_BSS_1.scr

Test Applet: API_2_PAH_FACRBBS_BSS_1.java

Load Script: API_2_PAH_FACRBBS_BSS_1.ldr

Cleanup Script: API_2_PAH_FACRBBS_BSS_1.clr

Parameter File: API_2_PAH_FACRBBS_BSS_1.par

6.2.7.17.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Initialise the handler	•	-
	findAndCompareValue() with a null compareBuffer	NullPointerException is thrown	
2	initDisplayText() with length = 15		
	<pre>compareOffset > compareBuffer.length tag = 0Dh, occurrence = 1 valueOffset = 0 compareBuffer.length = 5 compareOffset = 6 compareLength = 0</pre>	ArrayIndexOutOfBoundsException is thrown	
3	<pre>compareOffset < 0 compareBuffer.length = 5 compareOffset = -1 compareLength = 1</pre>	ArrayIndexOutOfBoundsException is thrown	
4	<pre>compareLength >compareBuffer.length compareBuffer.length = 5 compareOffset = 0 compareLength = 6</pre>	ArrayIndexOutOfBoundsException is thrown	
5	<pre>compareOffset + compareLength >compareBuffer.length compareBuffer.length = 5 compareOffset = 3 compareLength = 3</pre>	ArrayIndexOutOfBoundsException is thrown	
6	<pre>compareLength < 0 compareBuffer.length = 5 compareOffset = 0 compareLength = -1</pre>	ArrayIndexOutOfBoundsException is thrown	
7	initDisplayText() with length = 5		
	<pre>valueOffset > Text String Length tag = 0Dh, occurrence = 1 valueOffset = 7 compareBuffer.length = 15 compareOffset = 0 compareLength = 0</pre>	ToolkitException.OUT_OF_TLV_ BOUNDARIES is thrown	
8	<pre>valueOffset < 0 valueOffset = -1 compareBuffer.length = 15 compareOffset = 0 compareLength = 1</pre>	ToolkitException.OUT_OF_TLV_ BOUNDARIES is thrown	
9	<pre>compareLength > Text String length valueOffset = 0 compareBuffer.length = 15 compareOffset = 0 compareLength = 7</pre>	ToolkitException.OUT_OF_TLV_ BOUNDARIES is thrown	
10	<pre>valueOffset + compareLength > Text String</pre>	ToolkitException.OUT_OF_TLV_ BOUNDARIES is thrown	
11	Invalid parameter occurrence = 0	ToolkitException.BAD_INPUT_PA RAMETER is thrown	

ld	Description	API Expectation	APDU Expectation
12	InitDisplayText()	AFT Expediation	AF DO EXPECTATION
12	Select a TLV (tag 02h)		
	findAndCompareValue()	ToolkitException.UNAVAILABLE_	
	tag = 0Dh occurrence = 2	ELEMENT is thrown	
	Call the getValueLength() method	ToolkitException.UNAVAILABLE_ ELEMENT is thrown.	
13	initDisplayText()		
	dcs = 4 buffer = 00 01 OF		
	Initialise compareBuffer		
	compareBuffer =		
	04 00 01 OF		
	findAndCompareValue()	Result is 00h	
	tag = 0Dh, occurrence = 1 valueOffset = 0		
	compareOffset = 0		
	compareLength = 17		
14	Verify current TLV	Result is 17	
	getValueLength()		
15	Initialise compareBuffer		
'3	compareBuffer =		
	04 00 01 10		
	Compare buffers with same parameters	Result is -1	
16	Initialise compareBuffer		
	compareBuffer = 03 00 01 0F		
	Compare buffers with same parameters	Result is +1	
	Compare barrers with came parameters	Tresult is 11	
17	Initialise compareBuffer		
	compareBuffer =		
	55 55 55 01 02 03 04 05 06 07		
	08 09 0A 0B 0C		
	55 55 55 55		
	Compare buffers	Result is 00h	
	<pre>valueOffset = 2 compareOffset = 3</pre>		
	compareLength = 12		
18	Initialise compareBuffer		
	compareBuffer =		
	55 55 55 02 01 03 04 05 06 07		
	08 09 0A 0B 0C		
	55 55 55 55		
	Compare buffers with same parameters	Result is -1	
19	Initialise compareBuffer		
	compareBuffer = 55 55 55 01 02		
	03 04 05 06 07		
	08 09 0A 0A 0D		
	55 55 55 55 55 Compare buffers with some parameters	Deput in 14	
0.5	Compare buffers with same parameters	Result is +1	
20	append a Text String TLV tag = 0Dh		
	tag = 0Dn buffer = 00 11 22 33 44 55		
	Initialise compareBuffer		
	compareBuffer =		
	04 00 01 0F	Deput in OOk	
	<pre>findAndCompareValue() tag = 0Dh, occurrence = 1</pre>	Result is 00h	
	valueOffset = 0		
	compareOffset = 0		
	compareLength = 17		

ld	Description	API Expectation	APDU Expectation
21	Initialise compareBuffer		-
	compareBuffer =		
	00 11 22 33 44 55		
	findAndCompareValue()	Result is 00h	
	tag = 0Dh, occurrence = 2		
	valueOffset = 0		
	<pre>compareOffset = 0 compareLength = 6</pre>		
	compareneigth = 6		
22	Initialise compareBuffer		
	compareBuffer =		
	00 11 22 33 44 66		
	findAndCompareValue()	Result is -1	
	tag = 0Dh, occurrence = 2		
	valueOffset = 0		
	compareOffset = 0		
	compareLength = 6		
23	initDisplayText()		+
	dcs = 4		
	buffer = 00 01 0F		
	Initialise compareBuffer		
	CompareBuffer = 04 00 01 0F		
	Successful call (with tag 8Dh)	Result is 00h	
	tag = 8Dh, occurrence = 1		
	<pre>valueOffset = 0 compareBuffer.length = 17</pre>		
	compareOffset = 0		
	compareLength = 17		
24	Append tag 0Fh		
	buffer = 00 01 0F		
	Initialise compareBuffer		
	compareBuffer = 00 01 0F	D III aal	
	Successful call (with tag 8Fh)	Result is 00h	
	<pre>tag = 8Fh, occurrence = 1 valueOffset = 0</pre>		
	compareBuffer.length = 16		
	compareOffset = 0		
	compareLength = 16		
25	Initialise compareBuffer		
	compareBuffer =0099 02 OF		
	findAndCompareValue()	Result is +1	
	tag = 0Dh, occurrence = 1		
	valueOffset = 0		
	<pre>compareOffset = 0 compareLength = 17</pre>		
	COMPATENCIACII - II	1	

6.2.7.17.4 Test Coverage

CRR number	Test case number
N1	12
N2	14
N3	13, 17, 20, 21
N4	15, 18, 22
N5	16, 19
N6	23, 24
P1	1
P2	2, 3, 4, 5, 6
P3	7, 8, 9, 10
P4	11
C1	Does not apply for Proactive Handler

6.2.7.18 Method appendArray

Test Area Reference: API_2_PAH_APDA_BSS

6.2.7.18.1 Conformance requirement:

The method with following header shall be compliant to its definition in the API.

6.2.7.18.1.1 Normal execution

- CRRN1: appends a buffer into the Edithandler buffer.
- CRRN2: a successful append does not modify the TLV selected.

6.2.7.18.1.2 Parameters error

- CRRP1: if buffer is null, a java.lang.NullPointerException is thrown.
- CRRP2: if offset or length or both would cause access outside the array bounds, or if length is negative, a java.lang.ArrayIndexOutOfBoundsException is thrown.

6.2.7.18.1.3 Context errors

- CRRC1: if the EditHandler buffer is too small to append the requested data, a ToolkitException is thrown with reason code HANDLER_OVERFLOW.
- CRRC2: if the EditHandler buffer is busy, a ToolkitException is thrown with reason code HANDLER_NOT_AVAILABLE.

6.2.7.18.2 Test suite files

Test Script: API_2_PAH_APDA_BSS_1.scr
Test Applet: API_2_PAH_APDA_BSS_1.java
Load Script: API_2_PAH_APDA_BSS_1.ldr
Cleanup Script: API_2_PAH_APDA_BSS_1.clr
Parameter File: API_2_PAH_APDA_BSS_1.par

6.2.7.18.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Null buffer	NullPointerException is thrown	
2	offset > buffer.length	ArrayIndexOutOfBoundsExceptio	
	buffer.length = 5	n is thrown	
	offset = 6		
	length = 0		
3	offset < 0	ArrayIndexOutOfBoundsExceptio	
	buffer.length = 5	n is thrown	
	offset = -1		
	length = 1		
4	length > buffer.length	ArrayIndexOutOfBoundsExceptio	
	buffer.length = 5	n is thrown	
	offset = 0		
	length = 6		
5	offset + length > buffer.length	ArrayIndexOutOfBoundsExceptio	
	buffer.length = 5	n is thrown	
	offset = 3		
	length = 3		
6	length < 0	ArrayIndexOutOfBoundsExceptio	
	buffer.length = 5	n is thrown	
	offset = 0		
	length = -1		

ld	Description	API Expectation	APDU Expectation
7	Handler overflow	ToolkitException.HANDLER_OVE	·
	buffer.length = 256	RFLOW is thrown	
	offset = 0		
	length = 256		
8	Initialise handler		
	Select Command Details TLV		
	Successful call		
	buffer = FF FE F8		
	offset = 0		
	length = 8		
	Verify Current TLV: Call getValueLength()	Result is 03h	
9	Clear the handler		
	Successful call		
	buffer = FF FE F8		
	offset = 0		
	length = 8		
	Call copy() method		
	Compare the arrays	Result of	
	compareBuffer = FF FE F8	javacard.framework.Util.arrayCom	
		pare() is 00h	
10	Successful call		
	buffer = 00 01 07		
	offset = 2		
	length = 6		
	Call copy() method		
	Compare the arrays	Result of	
	compareBuffer = FF FE F8 02 03 07	javacard.framework.Util.arrayCom	
		pare() is 00h	
11	Successful call		
	buffer = 11 22 88		
	offset = 2		
	length = 4		
	Call copy() method		
	Compare the arrays	Result of	
	compareBuffer = FF FE F8 02 03 07 33	javacard.framework.Util.arrayCom	
	44 55 66	pare() is 00h	
12	Clear the handler		
	Successful call		
	buffer = 00 01 FC		
	offset = 0		
	length = 253		
	Call getLength() method	result = 253	
	Call copy() method		
	Compare handler	Result of	
	compareBuffer = 00 01 FC	javacard.framework.Util.arrayCom	
		pare() is 00h	

6.2.7.18.4 Test Coverage

CRR number	Test case number
N1	9, 10, 11, 12
N2	8
P1	1
P2	2, 3, 4, 5, 6
C1	7
C2	Does not apply for ProactiveHandler

6.2.7.19 Method appendTLV(byte tag, byte value)

Test Area Reference: API_2_PAH_APTLBB

6.2.7.19.1 Conformance requirement:

The method with following header shall be compliant to its definition in the API.

6.2.7.19.1.1 Normal execution

- CRRN1: Appends a TLV element to the current TLV list (1-byte element).
- CRRN2: A successful append does not modify the TLV selected.

6.2.7.19.1.2 Parameters error

No requirements

6.2.7.19.1.3 Context errors

- CRRC1: if the EditHandler buffer is too small to append the requested data, a ToolkitException is thrown with reason code HANDLER_OVERFLOW.
- CRRC2: if the EditHandler buffer is busy, a ToolkitException is thrown with reason code HANDLER_NOT_AVAILABLE.

6.2.7.19.2 Test suite files

Test Script: API_2_PAH_APTLBB_1.scr
Test Applet: API_2_PAH_APTLBB_1.java

Load Script: API_2_PAH_APTLBB_1.ldr

Cleanup Script: API_2_PAH_APTLBB_1.clr

Parameter File: API_2_PAH_APTLBB_1.par

6.2.7.19.3 Test procedure

ld	Description	ADI Expectation	APDU Expectation
	Description	API Expectation	APDO Expectation
1	Call appendArray()		
	length = 251		
	Handler Overflow: Call the appendTLV()	ToolkitException.HANDLER_OVE	
	method	RFLOW is thrown	
2	Initialise handler		
	Select Command Details TLV		
	Call the appendTLV() method		
	Verify Current TLV: Call getValueLength()	Result is 03h	
3	Clear the handler		
	Successful call		
	tag = 84h		
	value = 00h		
	Call copy() method		
	Compare the arrays	Result of	
	compareBuffer = 84 01 00	javacard.framework.Util.arrayCom	
		pare() is 00h	
4	Successful call	l V	
	tag = 01h		
	value = FEh		
	Call copy() method		
	Compare the arrays	Result of	
	compareBuffer = 84 01 00 01 01 FE	javacard.framework.Util.arrayCom	
		pare() is 00h	

ld	Description	API Expectation	APDU Expectation
5	Clear the handler		
	Call appendArray() length = 250 buffer = 00 81 F7 03 04 F9 Successful call tag = 84h		
	value = 00h Call getLength() method Call copy() method	result = 253	
	Compare the array compareBuffer = 00 81 F7 03 04 F9 84 01	Result of javacard.framework.Util.arrayCompare() is 00h	

6.2.7.19.4 Test Coverage

CRR number	Test case number
N1	3, 4, 5
N2	2
C1	1
C2	Does not apply for Proactive Handler

6.2.7.20 Method appendTLV(byte tag, byte value1, byte value2)

Test Area Reference: API_2_PAH_APTLBBB

6.2.7.20.1 Conformance requirements:

The method with following header shall be compliant to its definition in the API.

6.2.7.20.1.1 Normal execution

- CRRN1: Appends a TLV element to the current TLV list (2-byte element).
- CRRN2: A successful append does not modify the TLV selected.

6.2.7.20.1.2 Parameters error

No requirements

6.2.7.20.1.3 Context errors

- CRRC1: if the EditHandler buffer is too small to append the requested data, a ToolkitException is thrown with reason code HANDLER_OVERFLOW.
- CRRC2: if the EditHandler buffer is busy, a ToolkitException is thrown with reason code HANDLER_NOT_AVAILABLE.

6.2.7.20.2 Test suite files

Test Script: API_2_PAH_APTLBBB_1.scr
Test Applet: API_2_PAH_APTLBBB_1.java
Load Script: API_2_PAH_APTLBBB_1.ldr

Cleanup Script: API_2_PAH_APTLBBB_1.clr

Parameter File: API_2_PAH_APTLBBB_1.par

6.2.7.20.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Call the initDisplayText()	·	•
	length = 250		
	Handler Overflow: Call the appendTLV()	ToolkitException.HANDLER_OVE	
	method	RFLOW is thrown	
2	Initialise handler		
	Select Command Details TLV		
	Call the appendTLV() method		
	Verify Current TLV: Call getValueLength()	Result is 03h	
3	Clear the handler		
	Successful call		
	tag = 84h		
	value1 = 00h		
	value2 = 01h		
	Call copy() method	Decult of	
	Compare the arrays compareBuffer = 84 02 00 01	Result of javacard.framework.Util.arrayCom	
	Comparebuller - 04 02 00 01	pare() is 00h	
4	Successful call		
	tag = 01h		
	value1 = FEh		
	value2 = FDh Call copy() method		
	Compare the arrays	Result of	
	compareBuffer = 84 02 00 01 01 02 FE FD	javacard.framework.Util.arrayCom	
	comparedurer - or oz oo or or oz re rb	pare() is 00h	
5	Clear the handler		
	Groun and manufacture		
	Call appendArray()		
	length = 249		
	buffer = 00 81 F6 03 04 F8		
	Successful call		
	tag = 84h value1 = 00h		
	value2 = 01h		
	Call getLength() method	result = 253	
	gyyyy		
	Call copy() method		
	Compare handler	Result of	
	compareBuffer = 00 81 F6 03 04 F8 84 02	javacard.framework.Util.arrayCom	
	00 01	pare() is 00h	

6.2.7.20.4 Test Coverage

CRR number	Test case number
N1	3, 4, 5
N2	2
C1	1
C2	Does not apply for Proactive Handler

6.2.7.21 Method appendTLV(byte tag, byte[] value, short valueoffset, short valuelength)

Test Area Reference: API_2_PAH_APTLB_BSS

6.2.7.21.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

6.2.7.21.1.1 Normal execution

- CRRN1: Appends a TLV element to the current TLV list (byte-array element).
- CRRN2: A successful append does not modify the TLV selected.

6.2.7.21.1.2 Parameters error

- CRRP1: if value is null, a java.lang.NullPointerException is thrown.
- CRRP2: if valueoffset or valuelength or both would cause access outside the array bounds, or if length is negative, a java.lang.ArrayIndexOutOfBoundsException is thrown.

6.2.7.21.1.3 Context errors

- CRRC1: if the EditHandler buffer is too small to append the requested data, a ToolkitException is thrown with reason code HANDLER_OVERFLOW.
- CRRC2: if the EditHandler buffer is busy, a ToolkitException is thrown with reason code HANDLER_NOT_AVAILABLE.
- CRRC3: if valuelength is greater than 255, a ToolkitException is thrown with reason code BAD INPUT PARAMETER.

6.2.7.21.2 Test suite files

Test Script: API_2_PAH_APTLB_BSS_1.scr
Test Applet: API_2_PAH_APTLB_BSS_1.java
Load Script: API_2_PAH_APTLB_BSS_1.ldr
Cleanup Script: API_2_PAH_APTLB_BSS_1.clr
Parameter File: API_2_PAH_APTLB_BSS_1.par

6.2.7.21.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Null value	NullPointerException is thrown	
2	valueOffset > value.length	ArrayIndexOutOfBoundsExceptio	
	value.length = 5	n is thrown	
	valueOffset = 6		
	valueLength = 0		
3	valueOffset < 0	ArrayIndexOutOfBoundsExceptio	
	value.length = 5	n is thrown	
	valueOffset = -1		
	valueLength = 1		
4	valueLength > value.length	ArrayIndexOutOfBoundsExceptio	
	value.length = 5	n is thrown	
	valueOffset = 0		
	valueLength = 6		

ld	Description	API Expectation	APDU Expectation
5	valueOffset + valueLength > value.length	ArrayIndexOutOfBoundsExceptio	= 0 =
-	value.length = 5	n is thrown	
	valueOffset = 3		
	valueLength = 3	1 1 2 205	
6	valueLength < 0	ArrayIndexOutOfBoundsExceptio	
	<pre>value.length = 5 valueOffset = 0</pre>	n is thrown	
	valueLength = -1		
7	Handler overflow	ToolkitException.HANDLER_OVE	
'	value.length = 254	RFI OW is thrown	
	valueOffset = 0		
	valueLength = 254		
8	Bad parameter	ToolkitException.BAD_INPUT_PA	
	value.length = 256	RAMETER is thrown	
	<pre>valueOffset = 0 valueLength = 256</pre>		
9	Initialise handler		
	Select Command Details TLV		
	Successful call		
	tag = 04		
	value = FF FE F8		
	valueOffset = 0		
	valueLength = 8		
	Verify Current TLV: Call getValueLength()	Result is 03h	
10	Clear the handler		
	Successful call		
	tag = 04 value = FF FE F8		
	<pre>value = FF FE F8 valueOffset = 0</pre>		
	valueLength = 8		
	Call copy() method		
	Compare the arrays	Result of	
	compareBuffer = 04 08 FF FE F8	javacard.framework.Util.arrayCom	
		pare() is 00h	
11	Successful call		
	tag = 85h		
	value = 00 01 07		
	<pre>valueOffset = 2 valueLength = 6</pre>		
	Call copy() method		
	Compare the arrays	Result of	
	compareBuffer = 04 08 FF FE F8 85 06 02	javacard.framework.Util.arrayCom	
	03 07	pare() is 00h	
12	Successful call	, ,	
-	tag = 01		
	value = 11 22 88		
	valueOffset = 2		
	valueLength = 4 Call copy() method		
	Call copy() method Compare the arrays	Result of	
	compare the arrays compareBuffer = 04 08 FF FE F8 85 06 02	javacard.framework.Util.arrayCom	
	03 07 01 04 33 44 55 66	pare() is 00h	
13	Clear the handler	paretti is our	
	Successful call		
	tag = 04		
	value = 00 01 7F		
	valueOffset = 0		
	valueLength = 80h		
	Call copy() method		
	Compare the arrays	Result of	
	compareBuffer = 04 81 80 00 017F	javacard.framework.Util.arrayCom	
		pare() is 00h	

ld	Description	API Expectation	APDU Expectation
14	Clear the handler		
	Successful call		
	tag = 04		
	value = 00 01 F9		
	valueOffset = 0		
	valueLength = 250		
	Call getLength() method	result = 253	
	Call copy() method		
	Compare handler	Result of	
	compareBuffer = 04 81 FA 00 01F9	javacard.framework.Util.arrayCom pare() is 00h	

6.2.7.21.4 Test Coverage

CRR number	Test case number
N1	10, 11, 12, 13, 14
N2	9
P1	1
P2	2, 3, 4, 5, 6
C1	7
C2	Does not apply for Proactive Handler
C3	8

6.2.7.22 Method appendTLV(byte tag, byte value1, byte[] value2, short value2offset, short value2length)

Test Area Reference: API_2_PAH_APTLBB_BSS

6.2.7.22.1 Conformance requirement:

The method with following header shall be compliant to its definition in the API.

6.2.7.22.1.1 Normal execution

- CRRN1: Appends a TLV element to the current TLV list (1 byte and a byte-array element).
- CRRN2: A successful append does not modify the TLV selected.

6.2.7.22.1.2 Parameters error

- CRRP1: if value2 is null, a java.lang.NullPointerException is thrown.
- CRRP2: if value2offset or value2length or both would cause access outside the array bounds, or if length is negative, a java.lang.ArrayIndexOutOfBoundsException is thrown.

6.2.7.22.1.3 Context errors

• CRRC1: if the EditHandler buffer is too small to append the requested data, a ToolkitException is thrown with reason code HANDLER_OVERFLOW.

- CRRC2: if the EditHandler buffer is busy, a ToolkitException is thrown with reason code HANDLER_NOT_AVAILABLE.
- CRRC3: if valuelength is greater than 255, a ToolkitException is thrown with reason code BAD_INPUT_PARAMETER.

6.2.7.22.2 Test suite files

Test Script: API_2_PAH_APTLBB_BSS_1.scr

Test Applet: API_2_PAH_APTLBB_BSS_1.java

Load Script: API_2_PAH_APTLBB_BSS_1.ldr

Cleanup Script: API_2_PAH_APTLBB_BSS_1.clr

Parameter File: API_2_PAH_APTLBB_BSS_1.par

6.2.7.22.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Null value2	NullPointerException is thrown	AF DO Expectation
2	value2Offset > value2.length	ArrayIndexOutOfBoundsExceptio	
	<pre>value2.length = 5 value20ffset = 6</pre>	n is thrown	
	<pre>value20ffset = 6 value2Length = 0</pre>		
3	value2Offset < 0	ArrayIndexOutOfBoundsExceptio	
3	value2.length = 5	n is thrown	
	value20ffset = -1	n is thrown	
	value2Length = 1		
4	value2Length > value2.length	ArrayIndexOutOfBoundsExceptio	
•	value2.length = 5	n is thrown	
	value2Offset = 0	ii io unowii	
	value2Length = 6		
5	value2Offset + value2Length > value2.length	ArrayIndexOutOfBoundsExceptio	
	value2.length = 5	n is thrown	
	value2Offset = 3		
	value2Length = 3		
6	value2Length < 0	ArrayIndexOutOfBoundsExceptio	
	value2.length = 5	n is thrown	
	value2Offset = 0		
	value2Length = -1		
7	Handler overflow	ToolkitException.HANDLER_OVE	
	value2.length = 254	RFLOW is thrown	
	value2Offset = 0		
	value2Length = 254	T HIS G DAD INDUT DA	
8	Bad parameter	ToolkitException.BAD_INPUT_PA	
	<pre>value2.length = 256 value20ffset = 0</pre>	RAMETER is thrown	
	value2Length = 256		
9	Initialise handler		
3	Select Command Details TLV		
	Successful call		
	tag = 04 value1 = 05		
	value2 = FF FE F8		
	value20ffset = 0		
	value2Length = 8		
	Verify Current TLV: Call getValueLength()	Result is 03h	
10	Clear the handler		
	Successful call		
	tag = 04		
	value1 = 05		
	value2 = FF FE F8		
	value2Offset = 0		
	value2Length = 8		
	Call copy() method		
	Compare the arrays	Result of	
	CompareBuffer = 04 09 05 FF FE F8	javacard.framework.Util.arrayCom	
		pare() is 00h	
	l	Pais() 10 0011	

ld	Description	API Expectation	APDU Expectation
11	Successful call	•	•
	tag = 85h		
	value1 = 55h		
	value2 = 00 01 07		
	value2Offset = 2		
	value2Length = 6		
	Call copy() method		
	Compare the arrays	Result of	
	compareBuffer =	javacard.framework.Util.arrayCom	
	04 09 05 FF FE F8	pare() is 00h	
40	85 07 55 02 03 07		
12	Successful call		
	tag = 01		
	value1 = 44h value2 = 11 22 88		
	value20ffset = 2		
	value2Length = 4		
	Call copy() method		
	Compare the arrays	Result of	
	CompareBuffer =	javacard.framework.Util.arrayCom	
	04 09 05 FF FE F8	pare() is 00h	
	85 07 55 02 03 07	pare() is our	
	01 05 44 33 44 55 66		
13	Clear the handler		
	Successful call		
	tag = 04		
	value1 = 00		
	value2 = 01 7F		
	value2Offset = 0		
	value2Length = 7Fh		
	Call copy() method	D 11 (
	Compare the arrays	Result of	
	compareBuffer = 04 81 80 00 017F	javacard.framework.Util.arrayCom	
		pare() is 00h	
14	Clear the handler		
-	One and the H		
	Successful call		
	tag = 04 value1 = 00		
	value2 = 00 F9		
	value2Offset = 0		
	value2Length = 249		
	Call getLength() method	result = 253	
	Call copy() method		
	Compare handler	Result of	
	compareBuffer = 04 81 FA 00 01F9	javacard.framework.Util.arrayCom	
		pare() is 00h	
		12410() 10 0011	

6.2.7.22.4 Test Coverage

CRR number	Test case number
N1	10, 11, 12, 13, 14
N2	9
P1	1
P2	2, 3, 4, 5, 6
C1	7
C2	Does not apply for Proactive Handler
C3	8

6.2.7.23 Method clear

Test Area Reference: API_2_PAH_CLER

6.2.7.23.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

6.2.7.23.1.1 Normal execution

• CRRN1: Clears the TLV list of an EditHandler

• CRRN2: Resets the current TLV selected.

6.2.7.23.1.2 Parameters error

No requirements

6.2.7.23.1.3 Context errors

• CRRC1: if the EditHandler buffer is busy, a ToolkitException is thrown with reason code HANDLER_NOT_AVAILABLE

6.2.7.23.2 Test suite files

Test Script: API_2_PAH_CLER_1.scr

Test Applet: API_2_PAH_CLER_1.java

Load Script: API_2_PAH_CLER_1.ldr

Cleanup Script: API_2_PAH_CLER_1.clr

Parameter File: API_2_PAH_CLER_1.par

6.2.7.23.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Initialise the handler	Result of getLength() is not null	
	Select Command Details TLV		
	Call the getLength() method		
	Clear the handler	Result of getLength() is 0	
	Call the getLength() method		
2	Call the getValueLength() method	ToolkitException.UNAVAILABLE_ELEMENT	
		is thrown	

6.2.7.23.4 Test Coverage

CRR number	Test case number
N1	1
N2	2
C1	Does not apply for Proactive Handler

6.2.7.24 Method getCapacity

Test Area Reference: API_2_PAH_GCAP

6.2.7.24.1 Conformance Requirement

The method with following header shall be compliant to its definition in the API.

public byte getCapacity()

6.2.7.24.1.1 Normal execution

• CRRN1: The method shall return the maximum size of the Simple TLV list managed by the handler.

6.2.7.24.2 Test suite files

Test Script: API_2_PAH_GCAP_1.scr

Test Applet: API_2_PAH_GCAP_1.java

Load Script: API_2_PAH_GCAP_1.ldr

Cleanup Script: API_2_PAH_GCAP_1.clr

Parameter File: API_2_PAH_GCAP_1.par

6.2.7.24.3 Test Procedure

ld	Description	API Expectation	APDU Expectation
1	ProactiveHandler available		
Ī	1- Send envelope SMS-PP Formatted	1- Applet is triggered	
	2- The applet calls getTheHandler()	2- No exception is thrown	
	3- The applet calls getCapacity() on the	3- No exception is thrown, the	
	ProactiveHandler	capacity shall not be null	
	4- The applet fills the handler with the maximum	4- No exception is thrown	
	capacity, using appendTLV() method		
	5- The applet calls clear() on the proactive handler	5- No exception is thrown	
	6- The applet fills the handler with the maximum		
	capacity plus one, using appendTLV() method	6- HANDLER_OVERFLOW	
		exception is thrown	

6.2.7.24.4 Test Coverage

CRR number	Test case number	
N1	1	

6.2.7.25 Method initCloseChannel

Test Area Reference: API_2_PAH_ICCHB

6.2.7.25.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

public void initCloseChannel(byte bChannelIdentifier)

6.2.7.25.1.1 Normal execution

- CRRN1: The method shall build a Close Channel Proactive command, using Channel Identifier. Comprehension Required flags are set.
- CRRN2: A call to this method clears the handler then initialises it with Close Channel Proactive command.
- CRRN3: After the method invocation, no TLV is selected.
- CRRN4: The Close Channel Proactive command is not sent by the method.

6.2.7.25.2 Test suite files

Test Script: API_2_PAH_ICCHB_1.scr

Test Applet: API_2_PAH_ICCHB_1.java

Load Script: API_2_PAH_ICCHB_1.ldr

Cleanup Script: API_2_PAH_ICCHB_1.clr

Parameter File: API_2_PAH_ICCHB_1.par

6.2.7.25.3 Test procedure

ld	Description	API Expectation	APDU Expectation
0	Applet1 is installed with maximum number of channel = 01.		
1	Call initCloseChannel() method 1- Call ProactiveHandler.init() method to	2- Applet1 is triggered.	1- OPEN CHANNEL proactive command is fetched.
	Open a Channel. Call the ProactiveHandler.send() method. 2- Send an EVENT_EVENT_DOWNLOAD_CHANNEL_STATUS Envelope.	5- Applet1 is not triggered.	TERMINAL RESPONSE of OPEN CHANNEL is sent to the SIM with Channel Id = 01.
	3- Call the ProactiveHandler.initCloseChannel() method with Channel Id = 01. 4- Call the ProactiveHandler.send() method.		4- CLOSE CHANNEL proactive command is fetched. TERMINAL RESPONSE of
	5- Send an EVENT_EVENT_DOWNLOAD_CHANNEL_STATUS Envelope.		CLOSE CHANNEL is sent to the SIM.
2	Call the initCloseChannel () method with any value then build and send a CLOSE CHANNEL command	5- Applet1 is not triggered.	1- OPEN CHANNEL proactive command is fetched.
	1- Call ProactiveHandler.init() to Open a Channel and ProactiveHandler.send() methods.		TERMINAL RESPONSE of OPEN CHANNEL is sent to the SIM with Channel Id = 01.
	2- ProactiveHandler.initCloseChannel() with Channel Id = 2		
	<pre>3- ProactiveHandler.initCloseChannel() with the Channel Id = 1. 4- call the send() method.</pre>		4- CLOSE CHANNEL proactive command is fetched.
	5- Send an EVENT_EVENT_DOWNLOAD_CHANNEL_STATUS Envelope.		TERMINAL RESPONSE of CLOSE CHANNEL is sent to the SIM.
3	Select a TLV in the ProactiveHandler Call the initCloseChannel () method 1- Call ProactiveHandler.init() method to	3- UNAVAILABLE_ELEMENT ToolkitException is thrown by getValueLength() method.	1- OPEN CHANNEL proactive command is fetched.
	open a Channel and call the ProactiveHandler.send() method. Select 1st TLV of the Proactive Handler.		TERMINAL RESPONSE of OPEN CHANNEL is sent to the SIM with Channel Id = 01.
	ProactiveHandler.initCloseChannel() method with Channel Id = 01. 3- Call the ViewHandler.getValueLength() method.		4- CLOSE CHANNEL proactive command is fetched.
	4- Call ProactiveHandler.send() method.		TERMINAL RESPONSE of CLOSE CHANNEL is sent to the SIM.
4	Call the initCloseChannel() without sending the command	3- Applet1 is triggered.	1- OPEN CHANNEL proactive command is fetched.
	1- Call ProactiveHandler.init() method to		

open a Channel and call the ProactiveHandler.send() method.	TERMINAL RESPON OPEN CHANNEL is s	
2- Call the ProactiveHandler.initCloseChannel() method with Channel Id = 01 without	the SIM with Channel 01.	l ld =
ProactiveHandler.send(). 3- Send an EVENT_EVENT_DOWNLOAD_CHANNEL_STATUS Envelope.	No proactive commar shall be sent. Expecte status is '9000'	

6.2.7.25.4 Test Coverage

CRR number	Test case number	
N1	1	
N2	2	
N3	3	
N4	2, 4	

6.2.8 Class ProactiveResponseHandler

6.2.8.1 Method copyAdditionalInformation

Test Area Reference: API_2_PRH_CPAI_BSS

6.2.8.1.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

6.2.8.1.1.1 Normal execution

- CRRN1: The copyAdditionalInformation() method shall copy a part of the additional information field from Result TLV element in dstBuffer, using dstOffset and dstLength.
- CRRN2: dstBuffer shall only be modified from dstOffset to (dstOffset + dstLength 1) (included).
- CRRN3: The method returns (dstOffset + dstLength).
- CRRN4: If a Result TLV element is available, it becomes the TLV selected after a call to the method.
- CRRN5: The method shall copy from the first Result TLV.

6.2.8.1.1.2 Parameter errors

- CRRP1: A NullPointerException shall be thrown if dstBuffer is null.
- CRRP2: An ArrayIndexOutOfBoundsException shall be thrown if dstOffset or dstLength or both would cause
 access outside array bounds.

6.2.8.1.1.3 Context errors

CRRC1: A ToolkitException.UNAVAILABLE_ELEMENT shall be thrown in case of unavailable Result TLV element.

• CRRC2: A ToolkitException.OUT_OF_TLV_BOUNDARIES shall be thrown if dstLength is greater than the value field of the available TLV.

6.2.8.1.2 Test Suite files

Test Script: API_2_PRH_CPAI_BSS_1.scr

Test Applet: API_2_PRH_CPAI_BSS_1.java

Load Script: API_2_PRH_CPAI_BSS_1.ldr

Cleanup Script: API_2_PRH_CPAI_BSS_1.clr

Parameter File: API_2_PRH_CPAI_BSS_1.par

6.2.8.1.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Build and send a DISPLAY TEXT command		DISPLAY TEXT Proactive
	qualifier = 0		command
	dcs = 4		
	buffer = "Text"		
	Terminal Response with 11 additional bytes		
	Result TLV = 03 0C 01 01 23 45 67 89 AB		
	CD EF 01 23 45 NULL as parameter to dstBuffer	NullPointerException is thrown	
	dstBuffer = NULL	Indiii oiittei Exception is tillowii	
2	dstOffset > dstBuffer.length	ArrayIndexOutOfBoundsExceptio	
	dstBuffer.length = 10	n is thrown	
	dstOffset = 11		
	dstLength = 0		
3	dstOffset < 0	ArrayIndexOutOfBoundsExceptio	
	dstBuffer.length = 10	n is thrown	
	dstOffset = -1		
	dstLength = 1		
4	dstLength > dstBuffer.length	ArrayIndexOutOfBoundsExceptio	
	dstBuffer.length = 10	n is thrown	
	dstOffset = 0		
	dstLength = 11		
5	dstOffset + dstLength > dstBuffer.length	ArrayIndexOutOfBoundsExceptio	
	dstBuffer.length = 10	n is thrown	
	dstOffset = 6		
_	dstLength = 5	A manufactor to the country of the c	
6	dstLength < 0	ArrayIndexOutOfBoundsExceptio	
	<pre>dstBuffer.length = 10 dstOffset = 6</pre>	n is thrown	
	dstLength = -1		
7	Build and send a DISPLAY TEXT command		DISPLAY TEXT Proactive
,	Dulid and Selid a DISPEAT TEXT Command		command
	Terminal Response with 5 additional bytes		Command
	Result TLV = 03 06 01 01 23 45 67 89		
	Successful call, dstBuffer is the whole buffer	result of	
	dstBuffer.length = 5	copyAdditionalInformation() is	
	dstOffset = 0	05h.	
	dstLength = 5		
8	Compare dstBuffer using arrayCompare()	result of arrayCompare() is 00h.	
	src = {01, 23, 45, 67, 89}		
	srcOffset = 00		
	dest = dstBuffer		
	destOffset = 0		
9	length = 5 Call the getValueLength() method	Result is 06h.	
-			
10	Build and send a DISPLAY TEXT command		DISPLAY TEXT Proactive command
	Terminal Response with 6 additional bytes		
	1	1	

ld	Description	API Expectation	APDU Expectation
	Result TLV = 03 07 01 AB CD EF FE DC BA		
	Successful call, dstBuffer is part of a buffer dstBuffer.length = 7 dstOffset = 2 dstLength = 5	result of copyAdditionalInformation() is 07h.	
11	Compare dstBuffer using arrayCompare()	result of arrayCompare() is 00h.	
	<pre>src = {AB, CD, EF, FE, DC} srcOffset = 00 dest = dstBuffer destOffset = 2 length = 5</pre>		
12	Build and send a DISPLAY TEXT command		DISPLAY TEXT Proactive command
	Terminal Response with 7 additional bytes		
	Result TLV = 03 08 01 FE DC BA 98 76 54 32		
	Successful call, dstBuffer is part of a buffer dstBuffer.length = 7 dstOffset = 0 dstLength = 5	result of copyAdditionalInformation() is 05h.	
13	Compare dstBuffer using arrayCompare()	result of arrayCompare() is 00h.	
	<pre>src = {FE, DC, BA, 98, 76} srcOffset = 00 dest = dstBuffer destOffset = 0 length = 5</pre>		
14	Build and send a DISPLAY TEXT command		DISPLAY TEXT Proactive command
	Terminal Response with 8 additional bytes		
	Result TLV = 03 09 01 00 11 22 33 44 55 66 77		
	Successful call, dstBuffer is the whole buffer dstBuffer.length = 9 dstOffset = 2 dstLength = 5	result of copyAdditionalInformation() is 07h.	
15	Compare dstBuffer using arrayCompare()	result of arrayCompare() is 00h.	
	<pre>src = {00, 11, 22, 33, 44} srcOffset = 00 dest = dstBuffer destOffset = 2 length = 5</pre>		
16	Build and send a DISPLAY TEXT command		DISPLAY TEXT Proactive command
	Terminal Response with F2h additional bytes		
	Result TLV = 03 81 F3 01 00 01 02 03 Successful call to the method dstBuffer.length = F2h dstOffset = 0 dstLength = F2h	result of copyAdditionalInformation() is F2h.	
17	Compare dstBuffer using arrayCompare()	result of arrayCompare() is 00h.	
	<pre>src = {00, 01, 02, 03, 04} srcOffset = 00 dest = dstBuffer destOffset = 0 length = F2h</pre>		
18	Call the getValueLength() method	Result is F3h.	
19	Build and send a DISPLAY TEXT command		DISPLAY TEXT Proactive command
	Terminal Response with 5 additional bytes Result TLV = 03 06 01 00 11 22 33 44		
	dstLength > data available	OUT_OF_TLV_BOUNDARIES	
	dstBuffer.length = 6	SSI_SI_IEV_BOONDAINES	

ld	Description	API Expectation	APDU Expectation
	dstOffset = 0	ToolkitException is thrown	_
-00	dstLength = 6		DIODI AV TEVT D
20	Build and send a DISPLAY TEXT command		DISPLAY TEXT Proactive command
	Terminal Response with 5 additional bytes		Command
	Tommar Nooponeo mar o additional byteo		
	Result TLV = 03 06 01 00 11 22 33 44		
	Initialise dstBuffer dstBuffer = {00h, 01h, 02h, 03h}		
	Call the copyAdditionalInformation() method		
	dstBuffer.length = 20		
	dstOffset = 5 dstLength = 5		
	Compare dstBuffer using arrayCompare()	result of arrayCompare() is 00h	
	src = {		
	00h, 01h, 02h, 03h, 04h,		
	00h, 11h, 22h, 33h, 44h, 0Ah, 0Bh, 0Ch, 0Dh, 0Eh,		
	0Fh, 10h, 11h, 12h, 13h}		
	srcOffset = 0		
	dest = dstBuffer		
	destOffset = 0		
21	length = 20 Build and send a DISPLAY TEXT command		DISPLAY TEXT Proactive
21	Build and Send a Dioi EAT TEXT Command		command
	Terminal Response with 2 Result TLV		
	elements		
	1st Result TLV = 03 06 01 01 23 45 67 89 2nd Result TLV = 03 01 00		
	Successful call to copyAdditionalInformation()	result of	
	dstBuffer.length = 5	copyAdditionalInformation() is	
	dstOffset = 0	05h.	
	dstLength = 5		
22	Compare dstBuffer using arrayCompare()	result of arrayCompare() is 00h.	
	src = {01, 23, 45, 67, 89}		
	srcOffset = 00		
	dest = dstBuffer		
	destOffset = 0		
23	length = 5 Call the getValueLength() method	Result is 06h.	
20	Jan the gerralderengin() method	Troodit is oon.	
24	Build and send a DISPLAY TEXT command		DISPLAY TEXT Proactive
			command
	Terminal Response without Result Simple TLV	ToolkitException.UNAVAILABLE_	
		ELEMENT is thrown by send()	
	ProactiveResponseHandler, getTheHandler	ToolkitException.UNAVAILABLE_	
	call copyAdditionalInformation()	ELEMENT is thrown	

6.2.8.1.4 Test Coverage

CRR number	Test case number	
N1	8, 11, 13, 15, 17, 20, 22	
N2	20	
N3	7, 10, 12, 14, 16, 21	
N4	9, 18, 23	
N5	21, 22, 23	
P1	1	
P2	2, 3, 4, 5, 6	
C1	24	
C2	19	

6.2.8.2 Method copyTextString

Test Area Reference: API_2_PRH_CPTS_BS

6.2.8.2.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

6.2.8.2.1.1 Normal execution

- CRRN1: The copyTextString() method copies the text string value from the first Text String TLV element, using dstBuffer and dstOffset.
- CRRN2: If a Text String TLV element is available, it becomes the TLV selected.
- CRRN3: The method returns (dstOffset + length of copied value).

6.2.8.2.1.2 Parameter errors

- CRRP1: A NullPointerException shall be thrown if dstBuffer is null.
- CRRP2: A ArrayIndexOutOfBoundsException shall be thrown if dstOffset or dstOffset + (length of the TextString to be copied, without the Data Coding Scheme included), as specified for the returned value, would cause access outside array bounds.

6.2.8.2.1.3 Context errors

 CRRC1: A ToolkitException.UNAVAILABLE_ELEMENT shall be thrown in case of unavailable Text String TLV element.

6.2.8.2.2 Test Suite files

Test Script: API_2_PRH_CPTS_BS_1.scr

Test Applet: API_2_PRH_CPTS_BS_1.java

Load Script: API_2_PRH_CPTS_BS_1.ldr

Cleanup Script: API_2_PRH_CPTS_BS_1.clr

Parameter File: API_2_PRH_CPTS_BS_1.par

6.2.8.2.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Build and send a GET INPUT command		GET INPUT Proactive
	qualifier = 00h		command
	dcs = 04h		
	buffer = 'Text'		
	minRespLength = 00h		
	maxRespLength = FFh		
	Terminal Response		
	Text String TLV = 0D 02 04 41		
	ProactiveResponseHandler.getTheHandler();	NullPointerException is thrown	
	call the copyTextString() method with a null	·	
	dstBuffer		
	dstBuffer = null		
	dstOffset = 0		
2	Build and send a GET INPUT command		GET INPUT Proactive
			command

ld	Description	API Expectation	APDU Expectation
			Proactive
	Terminal Response		
	Text String TLV = 0D 04 04 "ABC"		
	dstOffset + text length > dstBuffer.length	ArrayIndexOutOfBoundsException	
	acconcer tom foregree accommend	is thrown	
	dstBuffer.length = 04h		
3	dstOffset = 02h dstOffset < 0	ArrayIndexOutOfBoundsException	
5	ustonisti v	is thrown	
	dstBuffer.length = 04h		
4	dstOffset = -1 Build and send a DISPLAY TEXT command		DISPLAY TEXT
4	qualifier = 00h		Proactive command
	dcs = 04h		Trodelive command
	buffer = 'Text'		
	Terminal Response without Text String TLV		
	ProactiveResponseHandler.getTheHandler();	UNAVAILABLE ELEMENT	
	call the copyTextString() method	ToolkitException is thrown	
5	Build and send a GET INPUT command		GET INPUT Proactive
			command Proactive
	Terminal Response with a null Text String TLV		Tioactive
	Tommar Roopones man a nam roke samig 124		
	Text String TLV = 0D 00		
	Initialise dstBuffer dstBuffer = {F00h, F01h, F02h, F03h}		
	Call the copyTextString() method	Result of copyTextString() is 02h	
	can and copy toxical mag () meaned	Treedit of copy toxicum g() to com	
	dstBuffer.length = 04h		
6	dstOffset = 02h Compare dstBuffer using arrayCompare()	Result of arrayCompare() is 00h	
0	Compare distribution distribution distribution	() is our	
	<pre>src = {0F0h, 0F1h, 0F2h, 0F3h}</pre>		
	<pre>srcOffset = 00h dest = dstBuffer</pre>		
	destOffset = 00h		
	length = 04h		
7	Build and send a GET INPUT command		GET INPUT Proactive command
			Proactive
	Terminal Response with text length = 01h		11000110
	Text String TLV = 0D 02 04 41		
	Initialise dstBuffer dstBuffer = {00h, 01h, 02h, 03h}		
	Call the copyTextString() method	Result of copyTextString() is 01h	
1	.,		
1	<pre>dstBuffer.length = 04h dstOffset = 00h</pre>		
8	Compare dstBuffer using arrayCompare()	Result of arrayCompare() is 00h	
1		,,,	
	<pre>src = {41h, 01h, 02h, 03h} srcOffset = 00h</pre>		
	dest = dstBuffer		
1	destOffset = 00h		
9	length = 04h Build and send a GET INPUT command		GET INPUT Proactive
ا	Duna ana sena a OET na OT command		command
L			Proactive
	Terminal Response with text length = 02h		
1	Text String TLV = 0D 03 04 42 43		
	Initialise dstBuffer		
	dstBuffer = {00h, 01h, 02h, 03h}		
	Call the copyTextString() method	Result of copyTextString() is 04h	
1	dstBuffer.length = 04h		
1	dstOffset = 02h		
10	Compare dstBuffer using arrayCompare()	Result of arrayCompare() is 00h	
		v	

ld	Description	API Expectation	APDU Expectation
	src = {00h, 01h, 42h, 43h}		
	srcOffset = 00h		
	<pre>dest = dstBuffer destOffset = 00h</pre>		
4.4	length = 04h	D. W. OOL	
11	Call the getValueLength() method	Result is 03h	
12	Build and send a GET INPUT command		GET INPUT Proactive
	Terminal Response with text length = 7Eh		command
	Text String TLV = 0D 7F 04 01 02 7E Initialise dstBuffer		
	dstBuffer = {00h, 00h 00h}		
	Call the copyTextString() method	Result of copyTextString() is 7Eh	
	dstBuffer.length = 7Eh dstOffset = 00h		
13	Compare dstBuffer using arrayCompare()	Result of arrayCompare() is 00h	
	src = {01h,, 7Eh}		
	srcOffset = 00h		
	<pre>dest = dstBuffer destOffset = 00h</pre>		
4.4	length = 7Eh Call the getValueLength() method	Deput is 75h	
14	Can the getvalueLength() method	Result is 7Fh	
15	Build and send a GET INPUT command		GET INPUT Proactive
	Terminal Response with text length = 7Fh		command
	-		
	Text String TLV = 0D 81 80 04 01 027F Initialise dstBuffer		
	dstBuffer = {00h, 01h FFh}	D + (T + (0) : (1) : (0.5)	
	Call the copyTextString() method	Result of copyTextString() is 8Fh	
	dstBuffer.length = FFh dstOffset = 10h		
16	Compare dstBuffer using arrayCompare()	Result of arrayCompare() is 00h	
	<pre>src = {00h, 01h, 0Fh,</pre>		
	01h,7Fh, 8Fh, FFh} srcOffset = 00h		
	dest = dstBuffer		
	<pre>destOffset = 00h length = FFh</pre>		
17	Build and send a GET INPUT command		GET INPUT Proactive
	Terminal Response with text length = EFh	<u> </u>	command
	Text String TLV = 0D 81 F0 04 01 02 EF Initialise dstBuffer		
	dstBuffer = {00h, 00h 00h}		
	Call the copyTextString() method	Result of copyTextString() is EFh	
	dstBuffer.length = FFh dstOffset = 00h		
18	Compare dstBuffer using arrayCompare()	Result of arrayCompare() is 00h	
	src = {01h,EFh, 00h 00h }	,	
	srcOffset = 00h		
	<pre>dest = dstBuffer destOffset = 00h</pre>		
40	length = FFh Build and send a GET INPUT command		OFT INDUT Describes
19	Dulia and Send a GET INPUT COMMAND		GET INPUT Proactive command
	Terminal Response with two Text String TLV		
	1st Text String TLV = 0D 03 04 42 43		
	2nd Text String TLV = 0D 02 04 44 Initialise dstBuffer		
	mindise asidallei		

ld	Description	API Expectation	APDU Expectation
	dstBuffer = {00h, 01h, 02h, 03h}		
	Call the copyTextString() method	Result of copyTextString() is 04h	
	dstBuffer.length = 04h dstOffset = 02h		
20	Compare dstBuffer using arrayCompare()	Result of arrayCompare() is 00h	
	<pre>src = {00h, 01h, 42h, 43h} srcOffset = 00h dest = dstBuffer destOffset = 00h length = 04h</pre>		
21	Call the getValueLength() method	Result is 03h	

6.2.8.2.4 Test Coverage

CRR number Test case numbe		
N1	6, 8, 10, 13, 16, 18, 20	
N2	11, 14, 21	
N3	5, 7, 9, 12, 15, 17, 19	
P1	1	
P2	2, 3	
C1	4	

6.2.8.3 Method getAdditionalInformationLength

Test Area Reference: API_2_PRH_GTIL

6.2.8.3.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

6.2.8.3.1.1 Normal execution

- CRRN1: This method returns the length of the additional information field from the first Result TLV in the ProactiveResponseHandler.
- CRRN2: After a successful execution of the method, the Result TLV becomes the selected TLV of the ProactiveResponseHandler.

6.2.8.3.1.2 Parameter errors

No requirements.

6.2.8.3.1.3 Context errors

CRRC1: A ToolkitException.UNAVAILABLE_ELEMENT shall be thrown in case of unavailable Result TLV element.

6.2.8.3.2 Test Suite files

Test Script: API_2_PRH_GTIL_1.scr

Test Applet: API_2_PRH_GTIL_1.java

Load Script: API_2_PRH_GTIL_1.ldr

Cleanup Script: API_2_PRH_GTIL_1.clr

Parameter File: API_2_PRH_GTIL_1.par

6.2.8.3.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Build and send a DISPLAY TEXT command	Arrespondin	DISPLAY TEXT Proactive
	qualifier = 00h		command
	dcs = 04h		
	buffer = 'Text' Terminal Response without additional		
	information		
	ProactiveResponseHandler.getTheHandler(); call the getAdditionalInformationLength() method	Result is 00h	
2	Call the getValueLength() method	Result is 01h	
3	Build and send a DISPLAY TEXT command		DISPLAY TEXT ProactiveProactive command
	Terminal Response with 1 additional byte		
	Result TLV = 03 02 02 55	D tri out	
	ProactiveResponseHandler.getTheHandler(); call the getAdditionalInformationLength() method	Result is 01h	
4	Call the getValueLength() method	Result is 02h	
5	Build and send a DISPLAY TEXT command		DISPLAY TEXT ProactiveProactive command
	Terminal Response with 7Eh additional bytes		
	Result TLV = 03 7F 02 55 55 55 ProactiveResponseHandler.getTheHandler();	Result is 7Eh	
	call the getAdditionalInformationLength() method	Result is 7EII	
6	Call the getValueLength() method	Result is 7Fh	
7	Build and send a DISPLAY TEXT command		DISPLAY TEXT Proactive command
	Terminal Response with 7Fh additional bytes		i rodelive esminana
	Result TLV = 03 81 80 02 55 55 55		
	ProactiveResponseHandler.getTheHandler(); call the getAdditionalInformationLength() method	Result is 7Fh	
8	Call the getValueLength() method	Result is 80h	
9	Build and send a DISPLAY TEXT command		DISPLAY TEXT Proactive command
	Terminal Response with 80h additional bytes Result TLV = 03 81 81 02 55 55 55		
	ProactiveResponseHandler.getTheHandler(); call the getAdditionalInformationLength() method	Result is 80h	
10	Call the getValueLength() method	Result is 81h	
11	Build and send a DISPLAY TEXT command		DISPLAY TEXT Proactive command
	Terminal Response with F2h additional bytes		
	Result TLV = 03 81 F3 02 55 55 55	Deput in E2h	
	ProactiveResponseHandler.getTheHandler(); call the getAdditionalInformationLength() method	Result is F2h	
12	Call the getValueLength() method	Result is F3h	

ld	Description	API Expectation	APDU Expectation
13	Build and send a DISPLAY TEXT command		DISPLAY TEXT
			Proactive command
	Terminal Response with 2 Result TLV		
	1st Result TLV = 03 03 02 01 23		
	2nd Result TLV = 03 01 00		
	ProactiveResponseHandler.getTheHandler();	Result is 02h	
	call the getAdditionalInformationLength() method		
14	Call the getValueLength() method	Result is 03h	
15	Build and send a DISPLAY TEXT command		DISPLAY TEXT
			Proactive command
	Terminal Response without Result Simple TLV	ToolkitException.UNAVAILABLE_E	
		LEMENT is thrown by send()	
	Get ProactiveResponseHandler		
	Call the getAdditionalInformationLength()	ToolkitException.UNAVAILABLE_E	
	method	LEMENT is thrown by	
		getAdditionalInformationLength ()	

6.2.8.3.4 Test Coverage

CRR number	Test case number	
N1	1, 3, 5, 7, 9, 11, 13	
N2	2, 4, 6, 8, 10, 12, 14	
C1	15	

6.2.8.4 Method getGeneralResult

Test Area Reference: API_2_PRH_GTGR

6.2.8.4.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

6.2.8.4.1.1 Normal execution

- CRRN1: This method returns the general result of a proactive command.
- CRRN2: After a successful execution of the method, the Result TLV becomes the selected TLV of the ProactiveResponseHandler.

6.2.8.4.1.2 Parameter errors

No requirements.

6.2.8.4.1.3 Context errors

- CRRC1: A ToolkitException.UNAVAILABLE_ELEMENT shall be thrown in case of unavailable Result TLV element.
- CRRC2: A ToolkitException.OUT_OF_TLV_BOUNDARIES shall be thrown if the general result byte is missing in the Result Simple TLV.

6.2.8.4.2 Test Suite files

Test Script: API_2_PRH_GTGR_1.scr

Test Applet: API_2_PRH_GTGR_1.java

Load Script: API_2_PRH_GTGR_1.ldr

Cleanup Script: API_2_PRH_GTGR_1.clr

Parameter File: API_2_PRH_GTGR_1.par

6.2.8.4.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Build and send a DISPLAY TEXT command	The Exposition	DISPLAY TEXT Proactive
	qualifier = 00h		command
	dcs = 04h buffer = 'Text'		
	Terminal Response with General Result = 00		
	(command performed successfully)		
	ProactiveResponseHandler.getTheHandler() Call the getGeneralResult() method	Result of getGeneralResult() is 00h	
2	Call the getValueLength() method	Result is 01h	
3	Build and send a DISPLAY TEXT command		DISPLAY TEXT Proactive command
	Terminal Response with General Result = 01, without Additional information on result (command performed with partial comprehension)		
	ProactiveResponseHandler.getTheHandler() Call the getGeneralResult() method	Result of getGeneralResult() is 01h	
4	Call the getValueLength() method	Result is 01h	
5	Build and send a DISPLAY TEXT command		DISPLAY TEXT Proactive command
	Terminal Response with General Result = 01, with Additional information on result Result TLV = 03 02 01 55 (command performed with partial comprehension)		
	ProactiveResponseHandler.getTheHandler() Call the getGeneralResult() method	Result of getGeneralResult() is 01h	
6	Call the getValueLength() method	Result is 02h	
7	Build and send a DISPLAY TEXT command		DISPLAY TEXT Proactive command
	Terminal Response with General Result = 02		
	Result TLV = 03 04 02 65 43 21 (Missing information)		
	ProactiveResponseHandler.getTheHandler() Call the getGeneralResult() method	Result of getGeneralResult() is 02h	
8	Call the getValueLength() method	Result is 04h	
9	Build and send a DISPLAY TEXT command		DISPLAY TEXT Proactive command
	Terminal Response with 7Fh additional bytes		
	ProactiveResponseHandler.getTheHandler(); call the getGeneralResult() method	Result is 02h	
10	Call the getValueLength() method	Result is 80h	

ld	Description	API Expectation	APDU Expectation
11	Build and send a DISPLAY TEXT command		DISPLAY TEXT Proactive command
	Terminal Response with 2 Result TLV		
	1st Result TLV = 03 02 02 12 2nd Result TLV = 03 03 03 34 56		
	<pre>ProactiveResponseHandler.getTheHandler() ; call the getGeneralResult() method</pre>	Result is 02h	
12	Call the getValueLength() method	Result is 02h	
13	Build and send a DISPLAY TEXT command		DISPLAY TEXT Proactive command
	Terminal Response without Result Simple TLV	ToolkitException.UNAVAILABLE_E LEMENT is thrown by send()	
	ProactiveResponseHandler.getTheHandler(); call the getGeneralResult() method	UNAVAILABLE_ELEMENT ToolkitException is thrown	
14	Build and send a DISPLAY TEXT command		DISPLAY TEXT Proactive command
	Terminal Response without General Result Byte in Result Simple TLV	ToolkitException.UNAVAILABLE_E LEMENT is thrown by send()	
	<pre>ProactiveResponseHandler.getTheHandler(); call the getGeneralResult() method</pre>	OUT_OF_TLV_BOUNDARIES ToolkitException is thrown	
	Result TLV = 03 00		

6.2.8.4.4 Test Coverage

CRR number	mber Test case number	
N1	1, 3, 5, 7, 9, 11	
N2	2, 4, 6, 8, 10, 12	
C1	13	
C2	14	

6.2.8.5 Method getItemIdentifier

Test Area Reference: API_2_PRH_GTII

6.2.8.5.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

6.2.8.5.1.1 Normal execution

- CRRN1: The method returns the item identifier byte value from the first Item Identifier TLV element.
- CRRN2: If an Item Identifier TLV element is available, it becomes the TLV selected.

6.2.8.5.1.2 Parameter errors

No requirements.

6.2.8.5.1.3 Context errors

- CRRC1: A ToolkitException.UNAVAILABLE_ELEMENT shall be thrown in case of unavailable Item Identifier TLV element.
- CRRC2: A ToolkitException.OUT_OF_TLV_BOUNDARIES shall be thrown if the item identifier byte is missing in the Item Identifier Simple TLV.

6.2.8.5.2 Test Suite files

Test Script: API_2_PRH_GTII_1.scr

Test Applet: API_2_PRH_GTII_1.java

Load Script: API_2_PRH_GTII_1.ldr

Cleanup Script: API_2_PRH_GTII_1.clr

Parameter File: API_2_PRH_GTII_1.par

6.2.8.5.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Build and send a DISPLAY TEXT command	7.1. 2.10000000	DISPLAY TEXT Proactive command
	Terminal Response (no Item Identifier TLV available)		
	Call to getItemIdentifier() with unavailable Item Identifier TLV	UNAVAILABLE_ELEMENT ToolkitException is thrown	
2	Build and send a SELECT ITEM command with 2 items (ID=01, 02)	·	SELECT ITEM Proactive command
	Terminal Response with Item 1 selected		
	Item Identifier TLV = 10 01 01 Call the getItemIdentifier() method	Result is 01h	
3	Call the getValueByte() method valueOffset = 00h	Result is 01h	
4	Build and send a SELECT ITEM command with 3 items (ID=03, 05, 07)		SELECT ITEM Proactive command
	Terminal Response with Item 5 selected		
	Item Identifier TLV = 10 01 05 Call the gettemIdentifier() method	Result is 05h	
5	Call the getValueByte() method valueOffset = 00h	Result is 05h	
6	Build and send a SELECT ITEM command with 3 items (ID=FDh, FEh, FFh)		SELECT ITEM Proactive command
	Terminal Response with Item FFh selected		
	Item Identifier TLV = 10 01 FF Call the getItemIdentifier() method	Result is FFh	
7	Call the getValueByte() method valueOffset = 00h	Result is FFh	
8	Build and send a SELECT ITEM command with 3 items (ID=FDh, FEh, FFh)		SELECT ITEM Proactive command
	Terminal Response with 2 Item Identifier TLV		
	1st Item Identifier TLV = 10 01 FFh 2nd Item Identifier TLV = 10 01 FEh		
	Call the getItemIdentifier() method	Result is FFh	

ld	Description	API Expectation	APDU Expectation
9	Call the getValueByte() method	Result is FFh	
	valueOffset = 00h		
10	Build and send a DISPLAY TEXT command		DISPLAY TEXT Proactive command
	Terminal Response without item identifier in the Item Identifier Simple TLV		
	Item Identifier TLV = 10 00		
	Call to getItemIdentifier()	OUT_OF_TLV_BOUNDARIES	
		ToolkitException is thrown	

6.2.8.5.4 Test Coverage

CRR number	Test case number
N1	2, 4, 6, 8
N2	3, 5, 7, 9
C1	1
C2	10

6.2.8.6 Method getTextStringCodingScheme

Test Area Reference: API_2_PRH_GTCS

6.2.8.6.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

6.2.8.6.1.1 Normal execution

- CRRN1: This method returns the data coding scheme from the first Text String TLV element.
- CRRN2: If a Text String TLV element is available, it becomes the TLV selected.

6.2.8.6.1.2 Parameter errors

No requirements.

6.2.8.6.1.3 Context errors

- CRRC1: A ToolkitException.UNAVAILABLE_ELEMENT shall be thrown in case of unavailable Text String TLV element.
- CRRC2: A ToolkitException.OUT_OF_TLV_BOUNDARIES shall be thrown if the Text String TLV is present with a length of 0.

6.2.8.6.2 Test Suite files

Test Script: API_2_PRH_GTCS_1.scr

Test Applet: API_2_PRH_GTCS_1.java

Load Script: API_2_PRH_GTCS_1.ldr

Cleanup Script: API_2_PRH_GTCS_1.clr

Parameter File: API_2_PRH_GTCS_1.par

6.2.8.6.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Build and send a DISPLAY TEXT command		DISPLAY TEXT Proactive command
	Terminal Response (no Text String TLV element available)		
	Call to getTextStringCodingScheme() with unavailable Text String TLV	UNAVAILABLE_ELEMENT ToolkitException is thrown	
2	Build and send a GET INPUT command		GET INPUT Proactive command
	Terminal Response with a null Text String TLV		
	Text String TLV = 0D 00 Call the getTextStringCodingScheme() method	OUT_OF_TLV_BOUNDARIES ToolkitException is thrown	
3	Build and send a GET INPUT command	Toolina Exception is thrown	GET INPUT Proactive command
	Terminal Response with text length = 01h, DCS = 04h		
	Text String TLV = 0D 02 04 "A"	 	
	Call the getTextStringCodingScheme() method	Result is 04h	
4	Call the getValueLength() method	Result is 02h	
5	Build and send a GET INPUT command		GET INPUT Proactive command
	Terminal Response with text length = 02h, DCS = 00h		
	Text String TLV = 0D 03 00 "BB"		
	Call the getTextStringCodingScheme() method	Result is 00h	
6	Call the getValueLength() method	Result is 03h	
7	Build and send a GET INPUT command		GET INPUT Proactive command
	Terminal Response with text length = 7Eh, DCS = 08h		
	Text String TLV = 0D 7F 08 01 02 7E		
	Call the getTextStringCodingScheme() method	Result is 08h	
8	Call the getValueLength() method	Result is 7Fh	
9	Build and send a GET INPUT command		GET INPUT Proactive command
	Terminal Response with text length = 7Fh, DCS = 04h		
	Text String TLV = 0D 81 80 04 01 02 7F		
	Call the getTextStringCodingScheme() method	Result is 04h	
10	Call the getValueLength() method	Result is 80h	

ld	Description	API Expectation	APDU Expectation
11	Build and send a GET INPUT command		GET INPUT Proactive command
	Terminal Response with text length = EFh, DCS = 08h		
	Text String TLV = 0D 81 F0 08 01 02 EE EF		
	Call the getTextStringCodingScheme() method	Result is 08h	
12	Call the getValueLength() method	Result is F0h	
13	Build and send a GET INPUT command		GET INPUT Proactive command
	Terminal Response with 2 Text String TLV		
	1st Text String TLV = 0D 02 04 41 2nd Text String TLV = 0D 03 08 42 43		
	Call the getTextStringCodingScheme() method	Result is 04h	
14	Call the getValueLength() method	Result is 02h	

6.2.8.6.4 Test Coverage

CRR number	Test case number
N1	3, 5, 7, 9, 11, 13
N2	4, 6, 8, 10, 12, 14
C1	1
C2	2

6.2.8.7 Method GetTextStringLength

Test Area Reference: API_2_PRH_GTTL

6.2.8.7.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

6.2.8.7.1.1 Normal execution

- CRRN1: The getTextStringLength() method returns the text string length value from the first Text String TLV
 element.
- CRRN2: If a Text String TLV element is available, it becomes the TLV selected.

6.2.8.7.1.2 Parameter errors

No requirements.

6.2.8.7.1.3 Context errors

• CRRC1: A ToolkitException.UNAVAILABLE_ELEMENT shall be thrown in case of unavailable Text String TLV element.

6.2.8.7.2 Test Suite files

Test Script: API_2_PRH_GTTL_1.scr

Test Applet: API_2_PRH_GTTL_1.java

Load Script: API_2_PRH_GTTL_1.ldr

Cleanup Script: API_2_PRH_GTTL_1.clr

Parameter File: API_2_PRH_GTTL_1.par

6.2.8.7.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Build and send a DISPLAY TEXT command		DISPLAY TEXT Proactive
			command
	Terminal Response (no Text String TLV element available)		
	Call to getTextStringLength() with unavailable Text String TLV	UNAVAILABLE_ELEMENT ToolkitException is thrown	
2	Build and send a GET INPUT command		GET INPUT Proactive command
	Terminal Response with a null Text String TLV Text String TLV = 0D 00		
	Call the getTextStringLength() method	Result is 00h	
3	Call the getValueLength() method	Result is 00h	
4	Build and send a GET INPUT command		GET INPUT Proactive command
	Terminal Response with text length = 01h, DCS = 04h Text String TLV = 0D 02 04 "A"		
	Call the getTextStringLength() method	Result is 01h	
5	Call the getValueLength() method	Result is 02h	
6	Build and send a GET INPUT command		GET INPUT Proactive command
	Terminal Response with text length = 02h, DCS = 00h		
	Text String TLV = 0D 03 00 "BB"		
	Call the getTextStringLength() method	Result is 02h	
7	Call the getValueLength() method	Result is 03h	
8	Build and send a GET INPUT command		GET INPUT Proactive command
	Terminal Response with text length = 7Eh, DCS = 08h Text String TLV = 0D 7F 08 01 02 7E		
	Call the getTextStringLength() method	Result is 7Eh	
9	Call the getValueLength() method	Result is 7Fh	
		•	

Build and send a GET INPUT command Terminal Response with text length = 7Fh, DCS = 04h Text String TLV = 0D 81 80 04 01 02 7F Call the getTextStringLength() method Result is 80h 12 Build and send a GET INPUT command Terminal Response with text length = EFh, DCS = 04h Text String TLV = 0D 81 F0 04 01 02 EE EF	1
Terminal Response with text length = 7Fh, DCS = 04h Text String TLV = 0D 81 80 04 01 02 7F Call the getTextStringLength() method Result is 7Fh 11 Call the getValueLength() method Result is 80h 12 Build and send a GET INPUT command GET INPUT Proactive command Terminal Response with text length = EFh, DCS = 04h Text String TLV = 0D 81 F0 04 01 02 EE	
DCS = 04h Text String TLV = 0D 81 80 04 01 02 7F Call the getTextStringLength() method Result is 7Fh 11 Call the getValueLength() method Result is 80h 12 Build and send a GET INPUT command GET INPUT Proactive command Terminal Response with text length = EFh, DCS = 04h Text String TLV = 0D 81 F0 04 01 02 EE	
DCS = 04h Text String TLV = 0D 81 80 04 01 02 7F Call the getTextStringLength() method Result is 7Fh 11 Call the getValueLength() method Result is 80h 12 Build and send a GET INPUT command GET INPUT Proactive command Terminal Response with text length = EFh, DCS = 04h Text String TLV = 0D 81 F0 04 01 02 EE	
Text String TLV = 0D 81 80 04 01 02 7F Call the getTextStringLength() method Result is 7Fh 11 Call the getValueLength() method Result is 80h 12 Build and send a GET INPUT command GET INPUT Proactive command Terminal Response with text length = EFh, DCS = 04h Text String TLV = 0D 81 F0 04 01 02 EE	
Call the getTextStringLength() method Result is 7Fh Call the getValueLength() method Result is 80h Build and send a GET INPUT command GET INPUT Proactive command Terminal Response with text length = EFh, DCS = 04h Text String TLV = 0D 81 F0 04 01 02 EE	
12 Build and send a GET INPUT command GET INPUT Proactive command Terminal Response with text length = EFh, DCS = 04h Text String TLV = 0D 81 F0 04 01 02 EE	
12 Build and send a GET INPUT command GET INPUT Proactive command Terminal Response with text length = EFh, DCS = 04h Text String TLV = 0D 81 F0 04 01 02 EE	
Terminal Response with text length = EFh, DCS = 04h Text String TLV = 0D 81 F0 04 01 02 EE	
Terminal Response with text length = EFh, DCS = 04h Text String TLV = 0D 81 F0 04 01 02 EE	
Terminal Response with text length = EFh, DCS = 04h Text String TLV = 0D 81 F0 04 01 02 EE	
DCS = 04h Text String TLV = 0D 81 F0 04 01 02 EE	
DCS = 04h Text String TLV = 0D 81 F0 04 01 02 EE	
Call the getTextStringLength() method Result is EFh	
13 Call the getValueLength() method Result is F0h	
ALL DELLA CONTROLLE DE LA CONT	
14 Build and send a GET INPUT command GET INPUT Proactive command	
Command	
Terminal Response with 2 Text String TLV	
1st Text String TLV = 0D 02 04 41	
2nd Text String TLV = 0D 03 08 42 43	
Call the getTextStringLength() method Result is 01h	
15 Call the getValueLength() method Result is 02h	

6.2.8.7.4 Test Coverage

CRR number	Test case number
1	2, 4, 6, 8, 10, 12, 14
2	3, 5, 7, 9, 11, 13, 15
3	1

6.2.8.8 Method getTheHandler

Test Area Reference: API_2_PRH_GTHD

6.2.8.8.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

 $\label{eq:public_public} \mbox{public static ProactiveResponseHandler getTheHandler()} \\ \mbox{throws ToolkitException}$

6.2.8.8.1.1 Normal execution

- CRRN1: The method shall return the single system instance of the ProactiveHandler class.
- CRRN2: The EnvelopeHandler is a Temporary JCRE Entry Point Object

6.2.8.8.1.2 Parameter errors

No requirements.

6.2.8.8.1.3 Context errors

• CRRC1: The method shall throw ToolkitException.HANDLER_NOT_AVAILABLE if the handler is busy.

6.2.8.8.2 Test Suite files

Test Script: API_2_PRH_GTHD_1.scr

Test Applet: API_2_PRH_GTHD_1.java

Load Script: API_2_PRH_GTHD_1.ldr

Cleanup Script: API_2_PRH_GTHD_1.clr

Parameter File: API_2_PRH_GTHD_1.par

6.2.8.8.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Build and send a Proactive Command		Proactive Command
	Terminal Response		
	getTheHandler() twice	The returned objects shall be the	
		same	
2	getTheHandler()	The reference shall be a	
		ProactiveResponseHandler	
3	getTheHandler()	The reference shall not be null	

6.2.8.8.4 Test Coverage

CRR number	Test case number
N1	1, 2, 3
N2	To be checked in Framework tests and insert here cross reference
C1	To be checked in Framework tests and insert here cross reference

6.2.8.9 Method getLength

Test Area Reference API_2_PRH_GLEN

6.2.8.9.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

6.2.8.9.1.1 Normal execution

• CRRN1: returns the length in bytes of the TLV list.

6.2.8.9.1.2 Parameter errors

No requirements.

6.2.8.9.1.3 Context errors

• CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException.HANDLER_NOT_AVAILABLE.

6.2.8.9.2 Test Suite files

Test Script: API_2_PRH_GLEN_1.scr

Test Applet: API_2_PRH_GLEN_1.java

Load Script: API_2_PRH_GLEN_1.ldr

Cleanup Script: API_2_PRH_GLEN_1.clr

Parameter File: API_2_PRH_GLEN_1.par

6.2.8.9.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Build and send a Display Text command		DISPLAY TEXT Proactive
			command
	Terminal Response without additional		
	information in General Result TLV		
	ProactiveResponseHandler.getTheHandler()	Result of getLength() is 12	
	getLength()		
2	Build and send a Display Text command		DISPLAY TEXT Proactive
			command
	Terminal Response with F2h additional		
	information in General Result TLV		
	ProactiveResponseHandler.getTheHandler()	Result of getLength() is FFh	
	getLength()		

6.2.8.9.4 Test Coverage

CRR number	Test case number	
N1	1, 2	
C1	Does not apply for Proactive Response Handler	

6.2.8.10 Method copy

Test Area Reference API_2_PRH_COPY_BSS

6.2.8.10.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

6.2.8.10.1.1 Normal execution

- CRRN1: copies the simple TLV list contained in the handler to the destination byte array.
- CRRN2: returns dstOffset + dstLength.

6.2.8.10.1.2 Parameter errors

- CRRP1: if dstBuffer is null a NullPointerException is thrown.
- CRRP2: if dstOffset or dstLength or both would cause access outside array bounds, or if dstLength is negative, an ArrayIndexOutOfBoundsException is thrown.

• CRRP3: if dstLength is grater than the length of the simple TLV List, an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException. OUT_OF_TLV_BOUNDARIES.

6.2.8.10.1.3 Context errors

• CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.

6.2.8.10.2 Test Suite files

Test Script: API_2_PRH_COPY_BSS_1.scr

Test Applet: API_2_PRH_COPY_BSS_1.java

Load Script: API_2_PRH_COPY_BSS_1.ldr

Cleanup Script: API_2_PRH_COPY_BSS_1.clr

Parameter File: API_2_PRH_COPY_BSS_1.par

6.2.8.10.3 Test procedure

	B	ADIE	1 ADDUE
ld	Description	API Expectation	APDU Expectation
1	Send a DISPLAY TEXT command		DISPLAY TEXT Proactive command
	Terminal Response without Additional		Command
	Information in General Result TLV:		
	81 03 01 21 00 02 02 82 81 03 01 00		
	ProactiveResponseHandler.getTheHandler()	NullPointerException is thrown	
	copy() with NULL as parameter to dstBuffer	·	
2	dstOffset > dstBuffer.length	ArrayIndexOutOfBoundsExceptio	
	dstBuffer.length = 5	n is thrown	
	dstOffset = 6		
	dstLength = 0		
3	dstOffset < 0	ArrayIndexOutOfBoundsExceptio	
	<pre>dstBuffer.length = 5 dstOffset = -1</pre>	n is thrown	
	dstUriset = -1 dstLength = 1		
4	dstLength > dstBuffer.length	ArrayIndexOutOfBoundsExceptio	
•	dstBuffer.length = 5	n is thrown	
	dstOffset = 0	ii io unowii	
	dstLength = 6		
5	dstOffset + dstLength > dstBuffer.length	ArrayIndexOutOfBoundsExceptio	
	dstBuffer.length = 5	n is thrown	
	dstOffset = 3		
6	dstLength = 3 dstLength < 0	ArrayIndexOutOfBoundsExceptio	
O	dstBuffer.length = 5	n is thrown	
	dstOffset = 0	II is unown	
	dstLength = -1		
7	dstLength > length of the simple TLV list	ToolkitException.OUT_OF_TLV_	
	dstBuffer.length = 13	BOUNDARIES is thrown	
	dstOffset = 0		
	dstLength = 13	D 1: (): 10	
8	Successful call, dstBuffer is the whole buffer	Result of copy() is 12	
	<pre>dstBuffer.length = 12 dstOffset = 0</pre>		
	dstLength = 12		
9	Compare the buffer with buffer:	Result of arrayCompare() is 0	
	81 03 01 21 00 02 02 82 81 03 01 00	result of array compare() to 0	
10	Successful call, dstBuffer is part of a buffer	Result of copy() is 15	
	dstBuffer.length = 20		
	dstOffset = 3		
	dstLength = 12		

ld	Description	API Expectation	APDU Expectation
11	Compare the whole buffer	Result of arrayCompare() is 0	
	Reference =		
	00 01 02		
	81 03 01 21 00		
	02 02 82 81		
	03 01 00		
	OF 10 11 12 13		
12	Initialise dstBuffer		
	dstBuffer = 00h 01h 02h 13h		
	Successful call, dstBuffer is part of a buffer	Result of copy() is 12	
	dstBuffer.length = 20		
	dstOffset = 3		
	dstLength = 9		
13	Compare the whole buffer	Result of arrayCompare() is 0	
	Reference =		
	00 01 02		
	81 03 01 21 00		
	02 02 82 81		
	OC OD OE		
	OF 10 11 12 13		

6.2.8.10.4 Test Coverage

CRR number	Test case number
N1	9, 11, 13
N2	8, 10, 12
P1	1
P2	2, 3, 4, 5, 6
P3	7
C1	Does not apply for Proactive Response Handler

6.2.8.11 Method findTLV

Test Area Reference API_2_PRH_FINDBB

6.2.8.11.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

6.2.8.11.1.1 Normal execution

Looks for the indicated occurrence of a TLV element from the beginning of the TLV list (handler buffer):

- CRRN1: the method is successful if the required occurrence exists then the corresponding TLV becomes current.
- CRRN2: if the method is successful then it returns TLV_FOUND_CR_SET when Comprehension Required flag is set.
- CRRN3: if the method is successful then it returns TLV_FOUND_CR_NOT_SET when Comprehension Required flag is not set.
- CRRN4: if the required occurrence of the TLV element does not exist, the current TLV is no longer defined and TLV_NOT_FOUND is returned.
- CRRN5: The search method is comprehension required flag independent.

6.2.8.11.1.2 Parameter errors

• CRRP1: if an input parameter is not valid (e.g. occurrence = 0) an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException BAD_INPUT_PARAMETER.

6.2.8.11.1.3 Context errors

• CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.

6.2.8.11.2 Test Suite files

Test Script: API_2_PRH_FINDBB_1.scr

Test Applet: API_2_PRH_FINDBB_1.java

Load Script: API_2_PRH_FINDBB_1.ldr

Cleanup Script: API_2_PRH_FINDBB_1.clr

Parameter File: API_2_PRH_FINDBB_1.par

6.2.8.11.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Send a DISPLAY TEXT command	AFI Expectation	DISPLAY TEXT Proactive
ı	Send a DISPLAT TEXT command		
	Terminal Beanance with 2 Canaral Beaut TIV		command
	Terminal Response with 2 General Result TLV 81 03 01 21 00		
	82 02 82 81		
	03 01 00		
	03 02 01 12		
	findTLV() with Invalid input parameter	ToolkitException.BAD_INPUT_PA	
	occurrence = 0	RAMETER is thrown	
2	Search 1st TLV	Result is TLV_FOUND_CR_SET	
	tag = 01h		
	occurrence = 1		
3	Call the getValueLength() method	Result is 03h	
4	Search 2nd TLV	Result is TLV_FOUND_CR_SET	
	tag = 02h		
	occurrence = 1		
5	Call the getValueLength() method	Result is 02h	
6	Select a TLV (tag 02h)		
	Search a wrong tag	Result is TLV_NOT_FOUND	
	tag = 04h		
7	occurrence = 1 Call the getValueLength() method	Tablitanantian UNIAN/AU ADI E	
/	Call the getvalueLength() method	ToolkitException.UNAVAILABLE_	
_	0	ELEMENT shall be thrown	
8	Search a tag with wrong occurrence tag = 01h	Result is TLV_NOT_FOUND	
	occurrence = 2		
9	Call the getValueLength() method	ToolkitException.UNAVAILABLE	
3	oan the getvaldezengin() method	ELEMENT shall be thrown.	
10	Search 3rd TLV	Result is	
10	tag = 03h	TLV_FOUND_CR_NOT_SET	
	occurrence = 1	TEV_1 00ND_OK_NO1_0E1	
11	Call the getValueLength() method	Result is 01h	
12	Search 3rd TLV	Result is	
	tag = 03h	TLV_FOUND_CR_NOT_SET	
	occurrence = 2		
13	Call the getValueLength() method	Result is 02h	
	-		
14	Search tag 83h	Result is	
	Tag = 83h	TLV_FOUND_CR_NOT_SET	
	Occurrence = 1		
15	Search tag 82h	Result is TLV_FOUND_CR_SET	
	Tag = 82h		
	Occurrence = 1		

6.2.8.11.4 Test Coverage

CRR number	Test case number
N1	3, 5, 11, 13
N2	2, 4
N3	10, 12
N4	6, 7,8, 9
N5	14,15
P1	1
C1	Does not apply for Proactive Response Handler

6.2.8.12 Method getValueLength

Test Area Reference API_2_PRH_GVLE

6.2.8.12.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

6.2.8.12.1.1 Normal execution

• CRRN1: gets and returns the binary length of the value field for the last TLV element which has been found in the handler.

6.2.8.12.1.2 Parameter errors

No requirements.

6.2.8.12.1.3 Context errors

- CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.
- CRRC2: in case of unavailable TLV element an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException UNAVAILABLE_ELEMENT.

6.2.8.12.2 Test Suite files

Test Script: API_2_PRH_GVLE_1.scr

Test Applet: API_2_PRH_GVLE_1.java

Load Script: API_2_PRH_GVLE_1.ldr

Cleanup Script: API_2_PRH_GVLE_1.clr

Parameter File: API_2_PRH_GVLE_1.par

6.2.8.12.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Send a GET INPUT command		GET INPUT Proactive
			command
	Terminal Response		
	Text String TLV = 0D 00		
	ProactiveResponseHandler.getTheHandler()	ToolkitException.UNAVAILABLE_	
	GetValueLength()	ELEMENT is thrown	
2	Search TLV 0Dh		
	getValueLength()	Result is 00h	

ld	Description	API Expectation	APDU Expectation
3	Send a GET INPUT command		GET INPUT Proactive
			command
	Terminal Response		
	Text String TLV = 0D 02 04 41		
	Search TLV 0Dh (Text String TLV)		
	getValueLength()	Result is 02h	
4	Send a GET INPUT command		GET INPUT Proactive command
	Terminal Response, Text String length = 7Eh		
	Text String TLV = 0D 7F 04 01 02 7E		
	Search TLV 0Dh (Text String TLV)		
	getValueLength()	Result is 7Fh	
5	Send a GET INPUT command		GET INPUT Proactive command
	Terminal Response, Text String length = 7Fh Text String TLV = 0D 81 80 04 01 02 7E 7F		
	Search TLV 0Dh (Text String TLV)		
	getValueLength()	Result is 80h	
6	Send a GET INPUT command		GET INPUT Proactive command
	Terminal Response, Text String length = EFh Text String TLV = 0D 81 F0 04 01 02 EF		
	Search TLV 0Dh (Text String TLV)		
	getValueLength()	Result is F0h	

6.2.8.12.4 Test Coverage

CRR number	Test case number
N1	2, 3, 4, 5, 6
C1	Does not apply for Proactive Response Handler
C2	1

6.2.8.13 Method getValueByte

Test Area Reference API_2_PRH_GVBYS

6.2.8.13.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

6.2.8.13.1.1 Normal execution

• CRRN1: Gets a byte from the last TLV element which has been found in the handler and returns its value (1 byte).

6.2.8.13.1.2 Parameter errors

• CRRP1: if valueOffset is out of the current TLV an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException OUT_OF_TLV_BOUNDARIES.

6.2.8.13.1.3 Context errors

• CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.

• CRRC2: in case of unavailable TLV element an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException UNAVAILABLE_ELEMENT.

6.2.8.13.2 Test Suite files

Test Script: API_2_PRH_GVBYS_1.scr

Test Applet: API_2_PRH_GVBYS_1.java

Load Script: API_2_PRH_GVBYS_1.ldr

Cleanup Script: API_2_PRH_GVBYS_1.clr

Parameter File: API_2_PRH_GVBYS_1.par

6.2.8.13.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Send a GET INPUT command		GET INPUT Proactive command
	Terminal Response, Text String length = 7Eh Text String TLV = 0D 7F 04 01 02 7E		
	ProactiveResponseHandler.getTheHandler()		
	getValueByte(0)	ToolkitException.UNAVAILABLE_ ELEMENT is thrown	
2	Search TLV 01h (Command Details TLV)		
	getValueByte(3)	ToolkitException.OUT_OF_TLV_ BOUNDARIES is thrown	
3	Search TLV 01h (Command Details TLV)		
	getValueByte(2)	Result is 00h (qualifier)	
4	Search TLV 02h (Device Identities TLV)		
	getValueByte(0)	Result is 82h (Source)	
5	Search TLV 0Dh (Text String TLV)		
	getValueByte(7E)	Result is 7Eh	
6	Send a GET INPUT command		GET INPUT Proactive command
	Terminal Response, Text String length = EFh Text String TLV = 0D 81 F0 04 01 02 7E 7F EF		
	Search TLV 0Dh (Text String TLV)		
	getValueByte(7E)	Result is 7Eh	
7	GetValueByte(7F)	Result is 7Fh	
8	GetValueByte(EF)	Result is EFh	

6.2.8.13.4 Test Coverage

CRR number	Test case number
N1	3, 4, 5, 6, 7, 8
P1	2
C1	Does not apply for Proactive Response Handler
C2	1

6.2.8.14 Method copyValue

Test Area Reference API_2_PRH_CPYVS_BSS

6.2.8.14.1 Conformance requirement

The method with following header shall be compliant with its definition in the API.

6.2.8.14.1.1 Normal execution

- CRRN1: copies a part of the last TLV element which has been found, into a destination. buffer.
- CRRN2: returns dstOffset + dstLength.

6.2.8.14.1.2 Parameter errors

- CRRP1: if dstBuffer is null NullPointerException is thrown.
- CRRP2: if dstOffset or dstLength or both would cause access outside array bounds, or if dstLength is negative ArrayIndexOutOfBoundsException is thrown.
- CRRP3: if valueOffset, dstLength or both are out of the current TLV an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException OUT_OF_TLV_BOUNDARIES.

6.2.8.14.1.3 Context errors

- CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.
- CRRC2: in case of unavailable TLV element an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException UNAVAILABLE ELEMENT.

6.2.8.14.2 Test Suite files

Test Script: API_2_PRH_CPYVS_BSS_1.scr

Test Applet: API_2_PRH_CPYVS_BSS_1.java

Load Script: API_2_PRH_CPYVS_BSS_1.ldr

Cleanup Script: API_2_PRH_CPYVS_BSS_1.clr

Parameter File: API_2_PRH_CPYVS_BSS_1.par

6.2.8.14.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Send a GET INPUT command		GET INPUT Proactive
			command
	Terminal Response, Text String length = 5		
	Text String TLV = 0D 06 04 01 02 05		
	ProactiveResponseHandler.getTheHandler()		
	Select Text String TLV		
	CopyValue() with a null dstBuffer	NullPointerException is thrown	
2	dstOffset > dstBuffer.length	ArrayIndexOutOfBoundsExceptio	
	dstBuffer.length = 5	n is thrown	
	dstOffset = 6		
	dstLength = 0		
3	dstOffset < 0	ArrayIndexOutOfBoundsExceptio	
	dstBuffer.length = 5	n is thrown	
	dstOffset = -1		
	dstLength = 1		

ld	Description	API Expectation	APDU Expectation
4	dstLength >dstBuffer.length	ArrayIndexOutOfBoundsExceptio	
	dstBuffer.length = 5 dstOffset = 0	n is thrown	
	dstLength = 6		
5	dstOffset + dstLength >dstBuffer.length	ArrayIndexOutOfBoundsExceptio	
	dstBuffer.length = 5	n is thrown	
	dstOffset = 3 dstLength = 3		
6	dstLength < 0	ArrayIndexOutOfBoundsExceptio	
	dstBuffer.length = 5	n is thrown	
	<pre>dstOffset = 0 dstLength = -1</pre>		
	dschength = -1		
7	valueOffset > Text String Length	ToolkitException.OUT_OF_TLV_	
	valueOffset = 7	BOUNDARIES is thrown	
	dstBuffer.length = 15 dstOffset = 0		
	dstLength = 0		
8	valueOffset < 0	ToolkitException.OUT_OF_TLV_	
	valueOffset = -1	BOUNDARIES is thrown	
	<pre>dstBuffer.length = 15 dstOffset = 0</pre>		
	dstLength = 1		
9	dstLength > Text String length valueOffset = 0	ToolkitException.OUT_OF_TLV_	
	dstBuffer.length = 15	BOUNDARIES is thrown	
	dstOffset = 0		
40	dstLength = 7	T 11:15 (1 OUT OF TIV	
10	ValueOffset + dstLength > Text String length ValueOffset = 2	ToolkitException.OUT_OF_TLV_ BOUNDARIES is thrown	
	DstBuffer.length = 15	BOONDAKIES IS UIIOWII	
	DstOffset = 0		
	DstLength = 5		
11	Send a GET INPUT command		GET INPUT Proactive
11			GET INPUT Proactive command
11	Terminal Response, Text String length = 16		
11	Terminal Response, Text String length = 16 Text String TLV = 0D 11 04 00 01 0F		
11	Terminal Response, Text String length = 16	ToolkitException.UNAVAILABLE	
11	Terminal Response, Text String length = 16 Text String TLV = 0D 11 04 00 01 0F ProactiveResponseHandler.getTheHandler CopyValue()	ToolkitException.UNAVAILABLE_ ELEMENT is thrown	
11	Terminal Response, Text String length = 16 Text String TLV = 0D 11 04 00 01 0F ProactiveResponseHandler.getTheHandler CopyValue() Select Text String TLV	ELEMENT is thrown	
	Terminal Response, Text String length = 16 Text String TLV = 0D 11 04 00 01 0F ProactiveResponseHandler.getTheHandler CopyValue() Select Text String TLV Successful call		
	Terminal Response, Text String length = 16 Text String TLV = 0D 11 04 00 01 0F ProactiveResponseHandler.getTheHandler CopyValue() Select Text String TLV	ELEMENT is thrown	
	Terminal Response, Text String length = 16 Text String TLV = 0D 11 04 00 01 0F ProactiveResponseHandler.getTheHandler CopyValue() Select Text String TLV Successful call ValueOffset = 0 DstBuffer.length = 17 DstOffset = 0	ELEMENT is thrown	
12	Terminal Response, Text String length = 16 Text String TLV = 0D 11 04 00 01 0F ProactiveResponseHandler.getTheHandler CopyValue() Select Text String TLV Successful call ValueOffset = 0 DstBuffer.length = 17 DstOffset = 0 DstLength = 17	Result of copyValue() is 17	
	Terminal Response, Text String length = 16 Text String TLV = 0D 11 04 00 01 0F ProactiveResponseHandler.getTheHandler CopyValue() Select Text String TLV Successful call ValueOffset = 0 DstBuffer.length = 17 DstOffset = 0 DstLength = 17 Compare buffer	ELEMENT is thrown	
12	Terminal Response, Text String length = 16 Text String TLV = 0D 11 04 00 01 0F ProactiveResponseHandler.getTheHandler CopyValue() Select Text String TLV Successful call ValueOffset = 0 DstBuffer.length = 17 DstOffset = 0 DstLength = 17	Result of copyValue() is 17	
12	Terminal Response, Text String length = 16 Text String TLV = 0D 11 04 00 01 0F ProactiveResponseHandler.getTheHandler CopyValue() Select Text String TLV Successful call ValueOffset = 0 DstBuffer.length = 17 DstOffset = 0 DstLength = 17 Compare buffer Buffer = 04 00 01 0F initialise dstBuffer	Result of copyValue() is 17	
12	Terminal Response, Text String length = 16 Text String TLV = 0D 11 04 00 01 0F ProactiveResponseHandler.getTheHandler CopyValue() Select Text String TLV Successful call ValueOffset = 0 DstBuffer.length = 17 DstOffset = 0 DstLength = 17 Compare buffer Buffer = 04 00 01 0F initialise dstBuffer dstBuffer = 55 55 55	Result of copyValue() is 17 Result is 00h	
12	Terminal Response, Text String length = 16 Text String TLV = 0D 11 04 00 01 0F ProactiveResponseHandler.getTheHandler CopyValue() Select Text String TLV Successful call ValueOffset = 0 DstBuffer.length = 17 DstOffset = 0 DstLength = 17 Compare buffer Buffer = 04 00 01 0F initialise dstBuffer	Result of copyValue() is 17	
12	Terminal Response, Text String length = 16 Text String TLV = 0D 11 04 00 01 0F ProactiveResponseHandler.getTheHandler CopyValue() Select Text String TLV Successful call ValueOffset = 0 DstBuffer.length = 17 DstOffset = 0 DstLength = 17 Compare buffer Buffer = 04 00 01 0F initialise dstBuffer dstBuffer = 55 55 55 Successful call ValueOffset = 2 DstBuffer.length = 20	Result of copyValue() is 17 Result is 00h	
12	Terminal Response, Text String length = 16 Text String TLV = 0D 11 04 00 01 0F ProactiveResponseHandler.getTheHandler CopyValue() Select Text String TLV Successful call ValueOffset = 0 DstBuffer.length = 17 DstOffset = 0 DstLength = 17 Compare buffer Buffer = 04 00 01 0F initialise dstBuffer dstBuffer = 55 55 55 Successful call ValueOffset = 2 DstBuffer.length = 20 DstOffset = 3	Result of copyValue() is 17 Result is 00h	
13	Terminal Response, Text String length = 16 Text String TLV = 0D 11 04 00 01 0F ProactiveResponseHandler.getTheHandler CopyValue() Select Text String TLV Successful call ValueOffset = 0 DstBuffer.length = 17 DstOffset = 0 DstLength = 17 Compare buffer Buffer = 04 00 01 0F initialise dstBuffer dstBuffer = 55 55 55 Successful call ValueOffset = 2 DstBuffer.length = 20 DstOffset = 3 DstLength = 12	Result of copyValue() is 17 Result is 00h Result of copyValue() is 15	
12	Terminal Response, Text String length = 16 Text String TLV = 0D 11 04 00 01 0F ProactiveResponseHandler.getTheHandler CopyValue() Select Text String TLV Successful call ValueOffset = 0 DstBuffer.length = 17 DstOffset = 0 DstLength = 17 Compare buffer Buffer = 04 00 01 0F initialise dstBuffer dstBuffer = 55 55 55 Successful call ValueOffset = 2 DstBuffer.length = 20 DstOffset = 3	Result of copyValue() is 17 Result is 00h	
13	Terminal Response, Text String length = 16 Text String TLV = 0D 11 04 00 01 0F ProactiveResponseHandler.getTheHandler CopyValue() Select Text String TLV Successful call ValueOffset = 0 DstBuffer.length = 17 DstOffset = 0 DstLength = 17 Compare buffer Buffer = 04 00 01 0F initialise dstBuffer dstBuffer = 55 55 55 Successful call ValueOffset = 2 DstBuffer.length = 20 DstOffset = 3 DstLength = 12 Compare buffer Buffer = 55 55 01 02	Result of copyValue() is 17 Result is 00h Result of copyValue() is 15	
13	Terminal Response, Text String length = 16 Text String TLV = 0D 11 04 00 01 0F ProactiveResponseHandler.getTheHandler CopyValue() Select Text String TLV Successful call ValueOffset = 0 DstBuffer.length = 17 DstOffset = 0 DstLength = 17 Compare buffer Buffer = 04 00 01 0F initialise dstBuffer dstBuffer = 55 55 55 Successful call ValueOffset = 2 DstBuffer.length = 20 DstOffset = 3 DstLength = 12 Compare buffer Buffer = 55 55 51 02 03 04 05 06 07	Result of copyValue() is 17 Result is 00h Result of copyValue() is 15	_
13	Terminal Response, Text String length = 16 Text String TLV = 0D 11 04 00 01 0F ProactiveResponseHandler.getTheHandler CopyValue() Select Text String TLV Successful call ValueOffset = 0 DstBuffer.length = 17 DstOffset = 0 DstLength = 17 Compare buffer Buffer = 04 00 01 0F initialise dstBuffer dstBuffer = 55 55 55 Successful call ValueOffset = 2 DstBuffer.length = 20 DstOffset = 3 DstLength = 12 Compare buffer Buffer = 55 55 01 02	Result of copyValue() is 17 Result is 00h Result of copyValue() is 15	_

6.2.8.14.4 Test Coverage

CRR number	Test case number
N1	13, 15
N2	12, 14
P1	1

CRR number	Test case number
P2	2, 3, 4, 5, 6
P3	7, 8, 9, 10
C1	Does not apply for Proactive Response Handler
C2	11

6.2.8.15 Method compare Value

Test Area Reference API_2_PRH_CPRVS_BSS

6.2.8.15.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

6.2.8.15.1.1 Normal execution

Compares the last found TLV element with a buffer:

- CRRN1: returns 0 if identical.
- CRRN2: returns -1 if the first miscomparing byte in simple TLV List is less than that in compareBuffer.
- CRRN3: returns 1 if the first miscomparing byte in simple TLV List is greater than that in compareBuffer.

6.2.8.15.1.2 Parameter errors

- CRRP1: if compareBuffer is null NullPointerException shall be thrown.
- CRRP2: if compareOffset or compareLength or both would cause access outside array bounds, or if compareLength is negative ArrayIndexOutOfBoundsException shall be thrown.
- CRRP3: if valueOffset, dstLength or both are out of the current TLV an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException OUT_OF_TLV_BOUNDARIES.

6.2.8.15.1.3 Context errors

- CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.
- CRRC2: in case of unavailable TLV element an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException UNAVAILABLE_ELEMENT.

6.2.8.15.2 Test Suite files

Test Script: API_2_PRH_CPRVS_BSS_1.scr

Test Applet: API_2_PRH_CPRVS_BSS_1.java

Load Script: API_2_PRH_CPRVS_BSS_1.ldr

Cleanup Script: API_2_PRH_CPRVS_BSS_1.clr

Parameter File: API_2_PRH_CPRVS_BSS_1.par

6.2.8.15.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Send a GET INPUT command		GET INPUT Proactive
			command
	Terminal Response, Text String length = 5		
	Text String TLV = 0D 06 04 01 02 05		
	ProactiveResponseHandler.getTheHandler() Select Text String TLV		
	compareValue() with a null compareBuffer	NullPointerException is thrown	
	oomparovalus() with a nam comparobation	Train office Exception is thown	
2	compareOffset > compareBuffer.length	ArrayIndexOutOfBoundsExceptio	
	compareBuffer.length = 5	n is thrown	
	compareOffset = 6		
3	compareLength = 0 compareOffset < 0	ArrayIndexOutOfBoundsExceptio	
3	compareBuffer.length = 5	In is thrown	
	compareOffset = -1	II IS UIIOWII	
	compareLength = 1		
4	compareLength >compareBuffer.length	ArrayIndexOutOfBoundsExceptio	
	<pre>compareBuffer.length = 5 compareOffset = 0</pre>	n is thrown	
	compareOffset = 0 compareLength = 6		
5	compareOffset + compareLength	ArrayIndexOutOfBoundsExceptio	
	>compareBuffer.length	n is thrown	
	compareBuffer.length = 5		
	compareOffset = 3		
6	compareLength = 3	ArrayIndayOutOfPaundaEyeantia	
٥	<pre>compareLength < 0 compareBuffer.length = 5</pre>	ArrayIndexOutOfBoundsException is thrown	
	compareOffset = 0	IT IS UTOWIT	
	compareLength = -1		
L			
7	valueOffset > Text String Length	ToolkitException.OUT_OF_TLV_	
	<pre>valueOffset = 7 compareBuffer.length = 15</pre>	BOUNDARIES is thrown	
	compareOffset = 0		
L	compareLength = 0		
8	valueOffset < 0	ToolkitException.OUT_OF_TLV_	
	valueOffset = -1	BOUNDARIES is thrown	
	<pre>compareBuffer.length = 15 compareOffset = 0</pre>		
	compareLength = 1		
9	compareLength > Text String length	ToolkitException.OUT_OF_TLV_	
	valueOffset = 0	BOUNDARIES is thrown	
	compareBuffer.length = 15		
	<pre>compareOffset = 0 compareLength = 7</pre>		
10	valueOffset + compareLength > Text String	ToolkitException.OUT_OF_TLV_	
	length	BOUNDARIES is thrown	
	valueOffset = 2		
	<pre>compareBuffer.length = 15 compareOffset = 0</pre>		
	compareOffset = 0 compareLength = 5		
L			
11	Send a GET INPUT command		GET INPUT Proactive
			command
	Terminal Response, Text String length = 16		
	Text String TLV = 0D 11 04 00 01 0F ProactiveResponseHandler.getTheHandler()		
	CompareValue()	ToolkitException.UNAVAILABLE_	
		ELEMENT is thrown	
12	Select Text String TLV		
	Initialise compareBuffer		
	CompareBuffer =		
	04 00 01 0F	D It is and	
	Compare buffers	Result is 00h	
	<pre>ValueOffset = 0 CompareOffset = 0</pre>		
	CompareLength = 17		
		1	

ld	Description	API Expectation	APDU Expectation
13	Initialise compareBuffer		
	CompareBuffer =		
	04 00 01 10		
	Compare buffers with same parameters	Result is -1	
14	Initialise compareBuffer		
	CompareBuffer =		
	03 00 01 0F		
	Compare buffers with same parameters	Result is +1	
15	Initialise compareBuffer		
	CompareBuffer =		
	55 55 55 01 02		
	03 04 05 06 07 08 09 0A 0B 0C		
	155 55 55 55 55		
	Compare buffers	Result is 00h	
	ValueOffset = 2	Result is oon	
	CompareOffset = 3		
	CompareLength = 12		
10	1.70 P		
16	Initialise compareBuffer CompareBuffer =		
	CompareBuller		
	03 04 05 06 07		
	08 09 0A 0B 0C		
	55 55 55 55		
	Compare buffers with same parameters	Result is -1	
17	Initialise compareBuffer		
	CompareBuffer =		
	55 55 55 01 02		
	03 04 05 06 07		
	08 09 0A 0A 0D		
	55 55 55 55 55	Darukia 14	
	Compare buffers with same parameters	Result is +1	

6.2.8.15.4 Test Coverage

CRR number	Test case number
N1	12, 15
N2	13, 16
N3	14, 17
P1	1
P2	2, 3, 4, 5, 6
P3	7, 8, 9, 10
C1	Does not apply for Proactive Response Handler
C2	11

6.2.8.16 Method findAndCopyValue(byte tag, byte[] dstBuffer, short valueOffset)

Test Area Reference API_2_PRH_FACYB_BS

6.2.8.16.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

6.2.8.16.1.1 Normal execution

- CRRN1: looks for the first occurrence of a TLV element from the beginning of a TLV list and copy its value into a destination buffer.
- CRRN2: if no TLV element is found, the UNAVAILABLE_ELEMENT exception is thrown and the current TLV is no longer defined.
- CRRN3: if the method is successful then the corresponding TLV becomes current and dstOffset + length of the copied value is returned.
- CRRN4: The search method is comprehension required flag independent.

6.2.8.16.1.2 Parameter errors

- CRRP1: if dstBuffer is null NullPointerException shall be thrown.
- CRRP2: if dstOffset would cause access outside array bounds ArrayIndexOutOfBoundsException shall be thrown.

6.2.8.16.1.3 Context errors

• CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.

6.2.8.16.2 Test Suite files

Test Script: API_2_PRH_FACYB_BS_1.scr

Test Applet: API_2_PRH_FACYB_BS_1.java

Load Script: API_2_PRH_FACYB_BS_1.ldr

Cleanup Script: API_2_PRH_FACYB_BS_1.clr

Parameter File: API_2_PRH_FACYB_BS_1.par

6.2.8.16.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Send a GET INPUT command	·	GET INPUT Proactive
			command
	Terminal Response, Text String length = 15		
	Text String TLV = 0D 10 04 01 02 0F		
	ProactiveResponseHandler.getTheHandler()		
	FindAndCopyValue() with a null dstBuffer	NullPointerException is thrown	
2	dstOffset > dstBuffer.length	ArrayIndexOutOfBoundsExceptio	
	tag = 0Dh	n is thrown	
	dstBuffer.length = 20		
	dstOffset = 21		
3	dstOffset < 0	ArrayIndexOutOfBoundsExceptio	
	dstBuffer.length = 20	n is thrown	
	dstOffset = -1		
4	length > dstBuffer.length	ArrayIndexOutOfBoundsExceptio	
	dstBuffer.length = 15	n is thrown	
	dstOffset = 0		
5	dstOffset + length >dstBuffer.length	ArrayIndexOutOfBoundsExceptio	
	dstBuffer.length = 20	n is thrown	
	dstOffset = 5		

Send a GET INPUT command	ld	Description	API Expectation	APDU Expectation
Terminal Response, Text String length = 16 Text String TLV = 0D 11 04 00 01 0F ProactiveResponseHandler.getTheHandler() Select at TLV (tag 0Zh) findAndCopyValue() tag = 04h Call the getValueLength() method Call the getValueLength() method Tag = 0Dh DatBuffer , length = 17 Datboffset = 0 Compare buffer Buffer = 04 00 01 0F Datbuffert = 20 D	6		•	
Text String TiV = 00 11 04 00 01 0F				
Text String TiV = 00 11 04 00 01 0F		Terminal Response, Text String length = 16		
ProactiveResponseHandler.getTheHandler() Select a TLV (tag 02h)				
Select a TLV (tag 02h)				
findAndCopyValue() tag = 04h Call the getValueLength() method ToolkitException.UNAVAILABLE_ELEMENT is thrown. Tog = 0Dh DstBuffer.length = 17 DstOffset = 0 8				
Lag = 04h			ToolkitException.UNAVAILABLE	
ELEMENT is thrown. Result of findAndcopyValue() is 17				
ELEMENT is thrown. Result of findAndcopyValue() is 17		Call the getValueLength() method	ToolkitException UNAVAILABLE	
Tag		3. 0		
Tag = 00h	7	Successful call		
DstOffset = 0			. , ,	
Result is 00h Result is 00h		~		
Buffer = 04 00 01 0F				
9 initialise dstBuffer dstBuffer = 55 55 55 Successful call DstBuffer.length = 20 DstOffset = 2 10 Compare buffer Buffer = 55 55 0 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F 55 11 Send a GET INPUT command Terminal Response, with 2 Text String TLV 0D 11 04 00 01 0F 0D 02 04 41 ProactiveResponseHandler.getTheHandler() Tag = 0Dh DstBuffer : 0 Successful call Tag = 0Dh DstBuffer : 0 Successful call Tag = 0 4 00 01 0F Buffer = 04 00 01 0F Buffer = 04 00 01 0F ProactiveResponseHandler.getTheHandler() Send a GET INPUT command Terminal Response, with 2 Text String length = 16 Text String TLV = 0D 11 04 00 01 0F ProactiveResponseHandler.getTheHandler() Successful call (with tag 8Dh) Tag = 8Dh DstBuffer.length = 17 ProactiveResponseHandler.getTheHandler() Successful call (with tag 8Dh) Tag = 8Dh DstBuffer.length = 17	8		Result is 00h	
Successful call Successful		Buffer = 04 00 01 0F		
Successful call Data Dat		initialia a dat Dotton		
Successful call DstBuffer .length = 20 19 19 19 19 19 19 19 1	9			
DetBuffer .length = 20			Popult of find Andoon (Aluc/) is	
DstOffset = 2 10 Compare buffer Buffer = 55 55 04 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F 55 11 Send a GET INPUT command Terminal Response, with 2 Text String TLV 0D 11 04 00 01 0F 0D 02 04 41 ProactiveResponseHandler.getTheHandler() Successful call Tag = 0Dh DstBuffer .length = 17 DstOffset = 0 22 Compare buffer Buffer = 04 00 01 0F Buffer = 04 00 01 0F Terminal Response, Text String length = 16 Text String TLV = 0D 11 04 00 01 0F ProactiveResponseHandler.getTheHandler() Successful call (with tag 8Dh) Tag = 8Dh DstBuffer.length = 17 Successful call (with tag 8Dh) Tag = 8Dh DstBuffer.length = 17				
Buffer = 55 55 04 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F 55 04 00 01 02 00 0E 0F 55 04 00 01 0 00 0E 0F 55 0 00 0D 0			19	
Buffer = 55 55 04 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F 55 04 00 01 02 00 0E 0F 55 04 00 01 0 00 0E 0F 55 0 00 0D 0	10	Compare buffer	Result is 00h	
11 Send a GET INPUT command GET INPUT Proactive command	. •			
11 Send a GET INPUT command GET INPUT Proactive command		55 55 04 00 01		
11 Send a GET INPUT command Terminal Response, with 2 Text String TLV 0D 11 04 00 01 0F 0D 02 04 41 ProactiveResponseHandler.getTheHandler() Successful call Tag = 0Dh DstBuffer.length = 17 DstOffset = 0 12 Compare buffer Buffer = 04 00 01 0F Buffer = 04 00 01 0F Terminal Response, Text String length = 16 Text String TLV = 0D 11 04 00 01 0F ProactiveResponseHandler.getTheHandler() Successful call (with tag 8Dh) Tag = 8Dh DstBuffer.length = 17 Result is 00h GET INPUT Proactive command Result is 00h				
Terminal Response, with 2 Text String TLV OD 11 04 00 01 OF OD 02 04 41 ProactiveResponseHandler.getTheHandler() Successful call Tag = 0Dh DstBuffer.length = 17 Dstoffset = 0 Compare buffer Buffer = 04 00 01 OF Buffer = 04 00 01 OF Terminal Response, Text String length = 16 Text String TLV = 0D 11 04 00 01 OF ProactiveResponseHandler.getTheHandler() Successful call (with tag 8Dh) DstBuffer.length = 17 Result of findAndcopyValue() is GET INPUT Proactive command GET INPUT Proactive command Result of findAndcopyValue() is 17				
Terminal Response, with 2 Text String TLV OD 11 04 00 01 OF OD 02 04 41 ProactiveResponseHandler.getTheHandler() Successful call Tag = 0Dh DstBuffer.length = 17 DstOffset = 0 12 Compare buffer Buffer = 04 00 01 OF 13 Send a GET INPUT command Terminal Response, Text String length = 16 Text String TLV = 0D 11 04 00 01 OF ProactiveResponseHandler.getTheHandler() Successful call (with tag 8Dh) DstBuffer.length = 17		OC OD OE OF 55		
Terminal Response, with 2 Text String TLV 0D 11 04 00 01 0F 0D 02 04 41 ProactiveResponseHandler.getTheHandler() Successful call Tag = 0Dh DstBuffer.length = 17 DstOffset = 0 12 Compare buffer Buffer = 04 00 01 0F 13 Send a GET INPUT command Terminal Response, Text String length = 16 Text String TLV = 0D 11 04 00 01 0F ProactiveResponseHandler.getTheHandler() Successful call (with tag 8Dh) DstBuffer.length = 17	11	Send a GET INPLIT command		GET INPLIT Proactive
Terminal Response, with 2 Text String TLV 0D 11 04 00 01 0F 0D 02 04 41 ProactiveResponseHandler.getTheHandler() Successful call Tag = 0Dh DstBuffer.length = 17 DstOffset = 0 12 Compare buffer Buffer = 04 00 01 0F 13 Send a GET INPUT command Terminal Response, Text String length = 16 Text String TLV = 0D 11 04 00 01 0F ProactiveResponseHandler.getTheHandler() Successful call (with tag 8Dh) Tag = 8Dh DstBuffer.length = 17		ocha a och ma or commana		
OD 11 04 00 01 0F OD 02 04 41 ProactiveResponseHandler.getTheHandler() Successful call Tag = 0Dh		Terminal Response with 2 Text String TI V		Command
ProactiveResponseHandler.getTheHandler() Successful call Tag = 0Dh DstBuffer.length = 17 DstOffset = 0 12 Compare buffer Buffer = 04 00 01 0F 13 Send a GET INPUT command Terminal Response, Text String length = 16 Text String TLV = 0D 11 04 00 01 0F ProactiveResponseHandler.getTheHandler() Successful call (with tag 8Dh) Tag = 8Dh DstBuffer.length = 17				
Successful call Tag = 0Dh DstBuffer.length = 17 DstOffset = 0 12 Compare buffer Buffer = 04 00 01 0F 13 Send a GET INPUT command Terminal Response, Text String length = 16 Text String TLV = 0D 11 04 00 01 0F ProactiveResponseHandler.getTheHandler() Successful call (with tag 8Dh) Tag = 8Dh DstBuffer.length = 17 Result of findAndcopyValue() is 17 Result of findAndcopyValue() is 17 Result of findAndcopyValue() is 17		OD 02 04 41		
Tag = 0Dh DstBuffer.length = 17 DstOffset = 0 12		ProactiveResponseHandler.getTheHandler()		
Tag = 0Dh DstBuffer.length = 17 DstOffset = 0 12		Successful call	Result of findAndcopyValue() is	
DstOffset = 0 12				
12		3		
Buffer = 04 00 01 0F 13 Send a GET INPUT command Terminal Response, Text String length = 16 Text String TLV = 0D 11 04 00 01 0F ProactiveResponseHandler.getTheHandler() Successful call (with tag 8Dh) Tag = 8Dh DstBuffer.length = 17 RESUlt of findAndcopyValue() is 17	4.0		B 11: 001	
Send a GET INPUT command Terminal Response, Text String length = 16 Text String TLV = 0D 11 04 00 01 0F ProactiveResponseHandler.getTheHandler() Successful call (with tag 8Dh) Tag = 8Dh DstBuffer.length = 17 GET INPUT Proactive command Result of findAndcopyValue() is 17	12	• • • • • • • • • • • • • • • • • • •	Result is 00h	
Terminal Response, Text String length = 16 Text String TLV = 0D 11 04 00 01 0F ProactiveResponseHandler.getTheHandler() Successful call (with tag 8Dh) Tag = 8Dh DstBuffer.length = 17	12			CET INDIT Proportive
Terminal Response, Text String length = 16 Text String TLV = 0D 11 04 00 01 0F ProactiveResponseHandler.getTheHandler() Successful call (with tag 8Dh) Tag = 8Dh DstBuffer.length = 17 Result of findAndcopyValue() is 17	13	Senu a GET INFOT COMMINANO		
Text String TLV = 0D 11 04 00 01 0F ProactiveResponseHandler.getTheHandler() Successful call (with tag 8Dh) Tag = 8Dh DstBuffer.length = 17 Text String TLV = 0D 11 04 00 01 0F Result of findAndcopyValue() is 17		Terminal Pesnanse Text String length - 16		COMMINATIO
ProactiveResponseHandler.getTheHandler() Successful call (with tag 8Dh) Tag = 8Dh DstBuffer.length = 17 Result of findAndcopyValue() is 17				
Successful call (with tag 8Dh) Tag = 8Dh DstBuffer.length = 17 Result of findAndcopyValue() is 17			<u> </u>	
Tag = 8Dh DstBuffer.length = 17			Result of findAndconv\/alue() is	
DstBuffer.length = 17				
14 Compare buffer Result is 00h	14	Compare buffer	Result is 00h	
Buffer = 04 00 01 0F				

6.2.8.16.4 Test Coverage

CRR number	Test case number
N1	8, 10, 12
N2	6
N3	7, 9, 11
N4	13, 14
P1	1
P2	2, 3, 4, 5
C1	Does not apply for Proactive Response Handler

6.2.8.17 Method findAndCopyValue(byte tag, byte occurence, short valueOffset, byte[] dstBuffer, short dstOffset, short dstLength)

Test Area Reference API_2_PRH_FACYBBS_BSS

6.2.8.17.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

6.2.8.17.1.1 Normal execution

- CRRN1: looks for the indicated occurrence of a TLV element from the beginning of a TLV list and copy its value into a destination buffer.
- CRRN2: if no TLV element is found, the UNAVAILABLE_ELEMENT exception is thrown and the current TLV is no longer defined.
- CRRN3: if the method is successful then the corresponding TLV becomes current and dstOffset + dstLength is returned.
- CRRN4: The search method is comprehension required flag independent.

6.2.8.17.1.2 Parameter errors

- CRRP1: if dstBuffer is null NullPointerException shall be thrown.
- CRRP2: if dstOffset or dstLength or both would cause access outside array bounds, or if dstLength is negative ArrayIndexOutOfBoundsException shall be thrown.
- CRRP3: if valueOffset, dstLength or both are out of the current TLV an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException OUT_OF_TLV_BOUNDARIES.

6.2.8.17.1.3 Context errors

• CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.

6.2.8.17.2 Test Suite files

Test Script: API_2_PRH_FACYBBS_BSS_1.scr

Test Applet: API_2_PRH_FACYBBS_BSS_1.java

Load Script: API_2_PRH_FACYBBS_BSS_1.ldr

Cleanup Script: API_2_PRH_FACYBBS_BSS_1.clr

Parameter File: API_2_PRH_FACYBBS_BSS_1.par

6.2.8.17.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Send a GET INPUT command		GET INPUT Proactive
			command
	Terminal Response, Text String length = 15		
	Text String TLV = 0D 10 04 01 02 0F ProactiveResponseHandler.getTheHandler()		
	findAndCopyValue() with a null dstBuffer	NullPointerException is thrown	
	marandoopy value() with a num detauler	Train offici Exception is thrown	
2	dstOffset > dstBuffer.length	ArrayIndexOutOfBoundsExceptio	
	tag = 0Dh, occurrence = 1	n is thrown	
	<pre>valueOffset = 0 dstBuffer.length = 5</pre>		
	dstOffset = 6		
3	dstLength = 0 dstOffset < 0	A manufactory of Of David da Evra anti-	
3	dstBuffer.length = 5	ArrayIndexOutOfBoundsException is thrown	
	dstOffset = -1	ii is unown	
_	dstLength = 1	Amendada OutOfDanada Farada	
4	<pre>dstLength > dstBuffer.length dstBuffer.length = 5</pre>	ArrayIndexOutOfBoundsException is thrown	
	dstOffset = 0	II 13 UIIOWII	
	dstLength = 6	Amenda de OutOf	
5	<pre>dstOffset + dstLength > dstBuffer.length dstBuffer.length = 5</pre>	ArrayIndexOutOfBoundsException is thrown	
	dstOffset = 3	II IS UIIOWII	
	dstLength = 3		
6	<pre>dstLength < 0 dstBuffer.length = 5</pre>	ArrayIndexOutOfBoundsException is thrown	
	dstOffset = 0	II IS UIIOWII	
	dstLength = -1		
7	Send a GET INPUT command		CET INDUIT Drop of the
′	Send a GET INPUT command		GET INPUT Proactive command
	Terminal Response, Text String length = 5		- Communication
	Text String TLV = 0D 06 04 01 02 05		
	ProactiveResponseHandler.getTheHandler()	Tablist Counties OUT OF TIV	
	<pre>valueOffset > Text String Length tag = 0Dh, occurrence = 1</pre>	ToolkitException.OUT_OF_TLV_ BOUNDARIES is thrown	
	valueOffset = 7	BOONDAINEO IS UNIOWIT	
	<pre>dstBuffer.length = 15 dstOffset = 0</pre>		
	dstLength = 0		
8	valueOffset < 0	ToolkitException.OUT_OF_TLV_	
	<pre>valueOffset = -1 dstBuffer.length = 15</pre>	BOUNDARIES is thrown	
	dstOffset = 0		
	dstLength = 1		
9	dstLength > Text String length valueOffset = 0	ToolkitException.OUT_OF_TLV_ BOUNDARIES is thrown	
	dstBuffer.length = 15	BOUNDARIES IS thrown	
	dstOffset = 0		
10	dstLength = 7 valueOffset + dstLength > Text String length	ToolkitException.OUT_OF_TLV_	
'0	valueOffset = 2	BOUNDARIES is thrown	
	dstBuffer.length = 15		
	<pre>dstOffset = 0 dstLength = 5</pre>		
11	Send a GET INPUT command		GET INPUT Proactive
	Torminal December 2 Tout Office I level 10		command
	Terminal Response, Text String length = 16 Text String TLV = 0D 11 04 00 01 0F		
	ProactiveResponseHandler.getTheHandler()		
	Select a TLV (tag 02h)		
	findAndCopyValue()	ToolkitException.UNAVAILABLE_	
	tag = 0Dh occurrence = 2	ELEMENT is thrown	
	Call the getValueLength() method	ToolkitException.UNAVAILABLE_	
	3	ELEMENT is thrown.	
_	·	· · · · · · · · · · · · · · · · · · ·	· · · · · · · · · · · · · · · · · · ·

231

ld	Description	API Expectation	APDU Expectation
12	Successful call	Result of findAndCopyValue() is	,
	Tag = 0Dh, occurrence = 1	17	
	ValueOffset = 0		
	DstBuffer.length = 17 DstOffset = 0		
	DstLength = 17		
13	Compare buffer	Result is 00h	
	Buffer = 04 00 01 0F		
14	initialise dstBuffer		
'	dstBuffer = 55 55 55		
	Successful call	Result of findAndcopyValue() is	
	Tag = 0Dh, occurrence = 1	15	
	ValueOffset = 2		
	DstBuffer.length = 20 DstOffset = 3		
	DstLength = 12		
15	Compare buffer	Result is 00h	
	Buffer =		
	55 55 55 01 02 03 04 05 06 07		
	08 09 0A 0B 0C		
	55 55 55 55		
16	Send a GET INPUT command		GET INPUT Proactive
			command
	Terminal Response, with 2 Text String TLV		
	0D 11 04 00 01 02 0F 0D 06 00 11 22 33 44 55 (no specific DCS		
	byte)		
	ProactiveResponseHandler.getTheHandler()		
	Successful call	Result of findAndCopyValue() is	
	Tag = 0Dh, occurrence = 1	17	
	ValueOffset = 0 DstBuffer.length = 17		
	DstOffset = 0		
	DstLength = 17		
17	Compare buffer	Result is 00h	
	Buffer = 04 00 01 0F		
18	Successful call	Result of findAndCopyValue() is 6	
	Tag = 0Dh, occurrence = 2		
	ValueOffset = 0		
	DstBuffer.length = 6 DstOffset = 0		
	DstLength = 6		
19	Compare buffer	Result is 00h	
	Buffer = 00 11 22 33 44 55		
20	Send a GET INPUT command		CET INDI IT Properties
20	Senu a GET INFUT COMMINANU		GET INPUT Proactive command
	Terminal Response, Text String length = 16		
	Text String TLV = 0D 11 04 00 01 0F		
	ProactiveResponseHandler.getTheHandler()		
	Successful call (with tag 8Dh)	Result of findAndcopyValue() is	
	Tag = 8Dh, occurrence = 1	17	
	ValueOffset = 0 DstBuffer.length = 17		
	DstOffset = 0		
	DstLength = 17		
21	Compare buffer	Result is 00h	
	Buffer = 04 00 01 0F		
			l .

6.2.8.17.4 Test Coverage

CRR number	Test case number
N1	13, 15, 17, 19
N2	11
N3	12, 14, 16, 18
N4	20, 21

CRR number	Test case number
P1	1
P2	2, 3, 4, 5, 6
P3	7, 8, 9, 10
C1	Does not apply for Proactive Response Handler

6.2.8.18 Method findAndCompareValue(byte tag, byte[] compareBuffer, short compareOffset)

Test Area Reference API_2_PRH_FACRB_BS

6.2.8.18.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

6.2.8.18.1.1 Normal execution

Looks for the first occurrence of a TLV element from beginning of a TLV list and compare its value with a buffer:

- CRRN1: if no TLV element is found, the UNAVAILABLE_ELEMENT exception is thrown and the current TLV is no longer defined.
- CRRN2: if the method is successful then the corresponding TLV becomes current.
- CRRN3: if identical returns 0.
- CRRN4: if the first miscomparing byte in simple TLV is less than that in compareBuffer returns -1.
- CRRN5: if the first miscomparing byte in simple TLV is greater than that in compareBuffer returns 1.
- CRRN6: The search method is comprehension required flag independent.

6.2.8.18.1.2 Parameter errors

- CRRP1: if compareBuffer is null NullPointerException shall be thrown.
- CRRP2: if compareOffset would cause access outside array bounds ArrayIndexOutOfBoundsException shall be thrown.

6.2.8.18.1.3 Context errors

• CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.

6.2.8.18.2 Test Suite files

Test Script: API_2_PRH_FACRB_BS_1.scr
Test Applet: API_2_PRH_FACRB_BS_1.java
Load Script: API_2_PRH_FACRB_BS_1.ldr
Cleanup Script: API_2_PRH_FACRB_BS_1.clr
Parameter File: API_2_PRH_FACRB_BS_1.par

6.2.8.18.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Send a GET INPUT command	AFFEXPONITION	GET INPUT Proactive command
	Terminal Response, Text String length = 15		
	Text String TLV = 0D 10 04 01 02 0F ProactiveResponseHandler.getTheHandler()		
	FindAndCompareValue() with a null dstBuffer	NullPointerException is thrown	
	This may and compare variable, man a man action of	Train onto Exception to the wife	
2	<pre>compareOffset > compareBuffer.length tag = 0Dh compareBuffer.length = 20 compareOffset = 21</pre>	ArrayIndexOutOfBoundsExceptio n is thrown	
3	<pre>compareOffset < 0 compareBuffer.length = 20 compareOffset = -1</pre>	ArrayIndexOutOfBoundsExceptio n is thrown	
4	<pre>length > compareBuffer.length compareBuffer.length = 15 compareOffset = 0</pre>	ArrayIndexOutOfBoundsExceptio n is thrown	
5	<pre>CompareOffset + length > compareBuffer.length CompareBuffer.length = 20 CompareOffset = 5</pre>	ArrayIndexOutOfBoundsException is thrown	
6	Send a GET INPUT command		GET INPUT Proactive command
	Terminal Response, Text String length = 16 Text String TLV = 0D 11 04 00 01 0F		
	ProactiveResponseHandler.getTheHandler()		
	Select a TLV (tag 02h)		
	findAndCompareValue() tag = 04h	ToolkitException.UNAVAILABLE_ ELEMENT is thrown	
	Call the getValueLength() method	ToolkitException.UNAVAILABLE_ ELEMENT is thrown.	
7	Initialise compareBuffer CompareBuffer = 04 00 01 0F		
	Compare buffers Tag = 0Dh CompareOffset = 0	Result is 00h	
8	Verify current TLV GetValueLength()	Result is 17	
9	Initialise compareBuffer CompareBuffer = 04 00 01 10		
	Compare buffers with same parameters	Result is -1	
10	Initialise compareBuffer CompareBuffer = 03 00 01 0F		
	Compare buffers with same parameters	Result is +1	
11	Initialise compareBuffer CompareBuffer = 55 55 04 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F 55 Compare buffers	Result is 00h	
	CompareOffset = 2		

ld	Description	API Expectation	APDU Expectation
12	Send a GET INPUT command	-	GET INPUT Proactive
			command
	Terminal Response, with 2 Text String TLV		
	0D 11 04 00 01 0F		
	OD 06 00 11 22 33 44 55		
	ProactiveResponseHandler.getTheHandler()		
	Initialise compareBuffer		
	CompareBuffer =		
	55 55 04 00 01		
	02 03 04 05 06		
	07 08 09 0A 0B 0C 0D 0E 0F 55		
	Compare buffers	Result is 00h	_
	CompareOffset = 2	Vegair 19 0011	
	Compareorisec - Z		
13	Initialise compareBuffer	 	
. •	CompareBuffer =		
	55 55 04 01 01		
	02 03 04 05 06		
	07 08 09 0A 0B		
	OC OD OE OF 55		
	Compare buffers	Result is -1	
	CompareOffset = 2		
14	Initialise compareBuffer		
	CompareBuffer =		
	55 55 04 00 01		
	02 03 04 05 06		
	07 08 09 0A 0B		
	OC OD OD 10 55	 	
	Compare buffers	Result is +1	
	CompareOffset = 2		
15	Send a GET INPUT command		GET INPUT Proactive
			command
	Terminal Response, Text String length = 16		
	Text String TLV = 0D 11 04 00 01 0F		
	ProactiveResponseHandler.getTheHandler()		
	Initialise compareBuffer		
	CompareBuffer =		
	04 00 01 0F		
	Compare buffers (with tag 8Dh)	Result is 00h	
	Tag = 8Dh		
	CompareOffset = 0		

6.2.8.18.4 Test Coverage

CRR number	Test case number
N1	6
N2	8
N3	7, 11, 12
N4	9, 13
N5	10, 14
N6	15
P1	1
P2	2, 3, 4, 5
C1	Does not apply for Proactive Response Handler

6.2.8.19 Method findAndCompareValue(byte tag, byte occurence, short valueOffset, byte[] compareBuffer, short compareOffset, short compareLength)

Test Area Reference API_2_PRH_FACRBBS_BSS

6.2.8.19.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

6.2.8.19.1.1 Normal execution

Looks for the indicated occurrence of a TLV element from the beginning of a TLV list and compare its value with a buffer:

- CRRN1: if no TLV element is found, the UNAVAILABLE_ELEMENT exception is thrown and the current TLV is no longer defined.
- CRRN2: if the method is successful then the corresponding TLV becomes current.
- CRRN3: if identical 0 is returned.
- CRRN4: if the first miscomparing byte in simple TLV is less than that in compareBuffer -1 is returned.
- CRRN5: if the first miscomparing byte in simple TLV is greater than that in compareBuffer 1 is returned
- CRRN6: The search method is comprehension required flag independent.

6.2.8.19.1.2 Parameter errors

- CRRP1: if compareBuffer is null NullPointerException shall be thrown.
- CRRP2: if compareOffset or compareLength or both would cause access outside array bounds, or if compareLength is negative ArrayIndexOutOfBoundsException shall be thrown.
- CRRP3: if valueOffset, compareLength or both are out of the current TLV an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException OUT_OF_TLV_BOUNDARIES.
- CRRP4: if an input parameter is not valid (e.g. occurrence = 0) an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException BAD_INPUT_PARAMETER.

6.2.8.19.1.3 Context errors

• CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.

6.2.8.19.2 Test Suite files

Test Script: API_2_PRH_FACRBBS_BSS_1.scr

Test Applet: API_2_PRH_FACRBBS_BSS_1.java

Load Script: API_2_PRH_FACRBBS_BSS_1.ldr

Cleanup Script: API_2_PRH_FACRBBS_BSS_1.clr

Parameter File: API_2_PRH_FACRBBS_BSS_1.par

6.2.8.19.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Send a GET INPUT command		GET INPUT Proactive command
	Terminal Response, Text String length = 15 Text String TLV = 0D 10 04 01 02 0F		
	ProactiveResponseHandler.getTheHandler()		
	findAndCompareValue() with a null compareBuffer	NullPointerException is thrown	
2	<pre>compareOffset > compareBuffer.length tag = 0Dh, occurrence = 1 valueOffset = 0 compareBuffer.length = 5 compareOffset = 6 compareLength = 0</pre>	ArrayIndexOutOfBoundsException is thrown	
3	<pre>compareOffset < 0 compareBuffer.length = 5 compareOffset = -1 compareLength = 1</pre>	ArrayIndexOutOfBoundsExceptio n is thrown	
4	<pre>compareLength >compareBuffer.length compareBuffer.length = 5 compareOffset = 0 compareLength = 6</pre>	ArrayIndexOutOfBoundsExceptio n is thrown	
	<pre>CompareOffset + compareLength >compareBuffer.length CompareBuffer.length = 5 CompareOffset = 3 CompareLength = 3</pre>	ArrayIndexOutOfBoundsException is thrown	
6	<pre>compareLength < 0 compareBuffer.length = 5 compareOffset = 0 compareLength = -1</pre>	ArrayIndexOutOfBoundsException is thrown	
7	Send a GET INPUT command		GET INPUT Proactive command
	Terminal Response, Text String length = 5		
	Text String TLV = 0D 06 04 01 02 05		
	ProactiveResponseHandler.getTheHandler()	TableitEvantion OUT OF TIV	
	<pre>valueOffset ≥ Text String Length tag = 0Dh, occurrence = 1 valueOffset = 7 compareBuffer.length = 15 compareOffset = 0 compareLength = 0</pre>	ToolkitException.OUT_OF_TLV_ BOUNDARIES is thrown	
8	<pre>valueOffset < 0 valueOffset = -1 compareBuffer.length = 15 compareOffset = 0 compareLength = 1</pre>	ToolkitException.OUT_OF_TLV_ BOUNDARIES is thrown	
9	<pre>compareLength > Text String length valueOffset = 0 compareBuffer.length = 15 compareOffset = 0 compareLength = 7</pre>	ToolkitException.OUT_OF_TLV_ BOUNDARIES is thrown	
10	<pre>valueOffset + compareLength > Text String</pre>	ToolkitException.OUT_OF_TLV_BOUNDARIES is thrown	
11	Invalid parameter Occurrence = 0	ToolkitException.BAD_INPUT_PA RAMETER is thrown	

ld	Description	API Expectation	APDU Expectation
12	Send a GET INPUT command	AllExpediation	GET INPUT Proactive
			command
	Terminal Response, Text String length = 16		
	Text String TLV = 0D 11 04 00 01 0F		
	ProactiveResponseHandler.getTheHandler()		
	Select a TLV (tag 02h)	Tablide Constitution LINIA VALLABLE	
	findAndCompareValue() tag = 0Dh	ToolkitException.UNAVAILABLE_ ELEMENT is thrown	
	occurrence = 2	LEEWENT IS UITOWIT	
	Call the getValueLength() method	ToolkitException.UNAVAILABLE_ ELEMENT is thrown.	
13	Initialise compareBuffer CompareBuffer = 04 00 01 0F		
	<pre>findAndCompareValue() tag = 0Dh, occurrence = 1 valueOffset = 0 compareOffset = 0 compareLength = 17</pre>	Result is 00h	
14	Verify current TLV GetValueLength()	Result is 17	
15	Initialise compareBuffer compareBuffer = 04 00 01 10		
	Compare buffers with same parameters	Result is -1	
16	Initialise compareBuffer compareBuffer = 03 00 01 0F		
	Compare buffers with same parameters	Result is +1	
17	Initialise compareBuffer compareBuffer = 55 55 55 01 02 03 04 05 06 07 08 09 0A 0B 0C 55 55 55 55 55		
	Compare buffers valueOffset = 2 compareOffset = 3 compareLength = 12	Result is 00h	
18	Initialise compareBuffer		
	compareBuffer = 55 55 55 02 01 03 04 05 06 07 08 09 0A 0B 0C 55 55 55 55 55		
	Compare buffers with same parameters	Result is -1	
19	Initialise compareBuffer compareBuffer = 55 55 55 01 02 03 04 05 06 07 08 09 0A 0A 0D 55 55 55 55 55 Compare buffers with same parameters	Result is +1	
	•		

ld	Description	API Expectation	APDU Expectation
20	Send a GET INPUT command		GET INPUT Proactive
			command
	Terminal Response, with 2 Text String TLV		
	OD 11 04 00 01 OF		
	OD 06 00 11 22 33 44 55		
	ProactiveResponseHandler.getTheHandler()		
	Initialise compareBuffer		
	compareBuffer =		
	04 00 01 0F	D 11: 001	
	findAndCompareValue()	Result is 00h	
	<pre>tag = 0Dh, occurrence = 1 valueOffset = 0</pre>		
	compareOffset = 0		
	compareLength = 17		
21	Initialise compareBuffer		
	compareBuffer =		
	00 11 22 33 44 55		
	findAndCompareValue()	Result is 00h	
	tag = 0Dh, occurrence = 2		
	valueOffset = 0		
	compareOffset = 0		
	compareLength = 6		
22	Initialise compareBuffer		
	compareBuffer =		
	00 11 22 33 44 66		
	findAndCompareValue()	Result is -1	
	tag = 0Dh, occurrence = 2		
	valueOffset = 0		
	compareOffset = 0		
	compareLength = 6		
23	Send a GET INPUT command		GET INPUT Proactive
			command
	Terminal Response, Text String length = 16		
	Text String TLV = 0D 11 04 00 01 0F		
	ProactiveResponseHandler.getTheHandler()		
	Initialise compareBuffer		
	CompareBuffer =		
	04 00 01 OF		
	Compare buffers (with tag 8Dh)	Result is 00h	
	tag = 8Dh, occurrence = 1		
	valueOffset = 0		
	compareOffset = 0		
	compareLength = 17		

6.2.8.19.4 Test Coverage

CRR number	Test case number
N1	12
N2	14
N3	13, 17, 20, 21
N4	15, 18, 22
N5	16, 19
N6	23
P1	1
P2	2, 3, 4, 5, 6
P3	7, 8, 9, 10
P4	11
C1	Does not apply for Proactive Response Handler

6.2.8.20 Method getCapacity

Test Area Reference: API_2_PRH_GCAP

6.2.8.20.1 Conformance Requirement

The method with following header shall be compliant to its definition in the API.

public byte getCapacity()

6.2.8.20.1.1 Normal execution

• CRRN1: The method shall return the maximum size of the Simple TLV list managed by the handler.

6.2.8.20.2 Test suite files

Test Script: API_2_PRH_GCAP_1.scr

Test Applet: API_2_PRH_GCAP_1.java

Load Script: API_2_PRH_GCAP_1.ldr

Cleanup Script: API_2_PRH_GCAP_1.clr

Parameter File: API_2_PRH_GCAP_1.par

6.2.8.20.3 Test Procedure

ld	Description	API Expectation	APDU Expectation
1	ProactiveResponseHandler available		
		1- Applet is triggered4-No exception is thrown	2- 91 XX 3- The proactive command is fetched
	5- The applet calls method getLength() method	5- The Capacity result is greater or equal to getLength() result	

6.2.8.20.4 Test Coverage

CRR number	Test case number
N1	1

6.2.8.21 Method getChannelldentifier

Test Area Reference: API_2_PRH_GCID

6.2.8.21.1 Conformance Requirement:

The method with following header shall be compliant to its definition in the API.

6.2.8.21.1.1 Normal execution

- CRRN1:The method shall return the channel identifier byte value.
- CRRN2:The channel identifier byte value returned shall be from the first Channel status TLV element.
- CRRN3: If the element is available it becomes the currently selected TLV.

6.2.8.21.1.2 Context errors

• CRRC1: The method shall throw ToolkitException (UNAVAILABLE_ELEMENT) if the Channel status TLV is not present.

• CRRC2: The method shall throw ToolkitException (OUT_OF_TLV_BOUNDARIES) if the Simple TLV Channel Status length is equal to 0.

6.2.8.21.2 Test suite files

Test Script: API_2_PRH_GCID_1.scr

Test Applet: API_2_PRH_GCID_1.java

Load Script: API_2_PRH_GCID_1.ldr

Cleanup Script: API_2_PRH_GCID_1.clr

Parameter File: API_2_PRH_GCID_1.par

6.2.8.21.3 Test Procedure

ld	Description	API Expectation	APDU Expectation
0	Applet1 is installed with maximum number of channel = 01.		
1	Channel status TLV is not present		1- DISPLAY TEXT
	1- Build and send a DISPLAY TEXT command		Proactive command is fetched.
	2- Call ProactiveResponseHandler.getChannelIdentif ier() method.	2- UNAVAILABLE_ELEMENT ToolkitException is thrown	TERMINAL RESPONSE with no Channel status TLV available.
2	Channel status TLV with a length equal to 0 1- Build and send a OPEN CHANNEL proactive command		1- OPEN CHANNEL Proactive command is fetched.
	2- Call ProactiveResponseHandler.getChannelIdentifier() method.	2- OUT_OF_TLV_BOUNDARIES ToolkitException is thrown	TERMINAL RESPONSE with Channel status TLV length equal to 0.
3	Get channel identifier value	2- Returns 0x01	1- OPEN CHANNEL
	1- Call ProactiveHandler.init() and ProactiveHandler.send() methods to open a channel. 2- Call		Proactive Command is fetched. TERMINAL RESPONSE is issued with channel status
	ProactiveResponseHandler.getChannelIdentifier() method.		value = 0x8100.
	3- Call ProactiveHandler.initCloseChannel() and ProactiveHandler.send() methods.		
4	Get channel identifier value with 2 TLV	2- Returns 0x01	1- OPEN CHANNEL Proactive Command is
	1- Call ProactiveHandler.init()and ProactiveHandler.send() methods to open a		fetched.
	channel 2- Call		TERMINAL RESPONSE is issued with channel status value = 0x8100 and 0x8200.
	<pre>ProactiveResponseHandler.getChannelIdentif ier()</pre>		value – exerce and exezes.
	3- Call ProactiveHandler.initCloseChannel() and ProactiveHandler.send() methods.		
5	Channel status TLV is currently selected TLV	2- Returns 0x03	1- OPEN CHANNEL Proactive Command is
	1- Call ProactiveHandler.init() and ProactiveHandler.send() methods to open a channel.	3- Check getChannelIdentifier() =getValueByte(0)	fetched.
	ViewHandler.FindTLV with Device Identity Tag.		TERMINAL RESPONSE is issued with channel status value = 0x0305.
	<pre>2- Call ProactiveResponseHandler.getChannelIdentif ier() method.</pre>		
	3- Compare ProactiveResponseHandler.getChannelIdentif ier() and then ViewHandler.getValueByte(0) methods.		

6.2.8.21.4 Test Coverage

CRR number	Test case number
N1	3
N2	4
N3	5
C1	1

C2	2

6.2.8.22 Method copyChannelData

Test Area Reference: API_2_PRH_CCHD_BSS

6.2.8.22.1 Conformance Requirement:

The method with following header shall be compliant to its definition in the API.

6.2.8.22.1 Normal execution

- CRRN1: The method shall copy a part of the Channel data string field.
- CRRN2: The Channel data string field value returned shall be the first Channel data TLV element of the current response data field.
- CRRN3: If the element is available it becomes the currenly selected TLV.
- CRRN4: Returns dstOffset + dstLength.

6.2.8.22.2 Parameters error

- CRRP1: If dstBuffer is null, a NullPointerException is thrown.
- CRRP2: If dstOffset or dstLength parameter is negative an ArrayIndexOutOfBoundsException exception is thrown and no copy is performed.
- CRRP3: If dstOffset+dstLength is greater than dstBuffer.length, the length of the dstBuffer array an ArrayIndexOutOfBoundsException exception is thrown and no copy is performed.
- CRRP4: If dstLength is greater than the value field of the available TLV, a OUT_OF_TLV_BOUNDARIES ToolkitException is thrown.

6.2.8.22.3 Context errors

CRRC1: The method shall throw a UNAVAILABLE_ELEMENT ToolkitException if the Result TLV is not present.

6.2.8.22.2 Test suite files

Test Script: API_2_PRH_CCHD_BSS_1.scr
Test Applet: API_2_PRH_CCHD_BSS_1.java
Load Script: API_2_PRH_CCHD_BSS_1.ldr
Cleanup Script: API_2_PRH_CCHD_BSS_1.clr
Parameter File: API_2_PRH_CCHD_BSS_1.par

6.2.8.22.3 Test Procedure

ld	Description	API Expectation	APDU Expectation
0	1- Applet1 is installed with maximum number of channel = 01.		2- OPEN CHANNEL proactive command is fetched
	2- Applet1 builds proactive commands OPEN CHANNEL with init() method in order to open one channel.		TERMINAL RESPONSE is issued with Channel Id = 01
1	ProactiveHandler.send() method is called. CopyChannelData() with NULL dstBuffer		RECEIVE DATA Proactive
'	Build and send a RECEIVE DATA command	NullPointerException is thrown	command is fetched. TERMINAL RESPONSE with not empty Channel
	Call ProactiveResponseHandler.copyChannelData dstBuffer = NULL DstOffset = 0 DstLength = 1		Data TLV is issued.
2	CopyChannelData() with negative dstOffset 1- call init() method for the RECEIVE DATA proactive command. 2- call	2- an ArrayIndexOutOfBoundsException exception is thrown.	1- RECEIVE DATA proactive command is fetched. TERMINAL RESPONSE
	ProactiveResponseHandler.copyChannelData() DstBuffer.length = 6 DstOffset = -1 DstLength = 1	3- no copy is performed.	with 6 bytes avalaible ('Hello1')
3	3- check dstBuffer is empty. CopyChannelData() with negative dstLength	1- an	
	1- call ProactiveResponseHandler.copyChannelData() DstBuffer.length = 6 DstOffset = 0 DstLength = -1	ArrayIndexOutOfBoundsException exception is thrown. 2- no copy is performed.	
	2- check dstBuffer is empty.		
4		1- an ArrayIndexOutOfBoundsException exception is thrown.	
	with dstOffset+dstLength greater than dstBuffer.length. DstBuffer.length = 6 DstOffset = 5 DstLength = 2	2- no copy is performed.	
	2- check dstBuffer is empty.		
5	Call ProactiveResponseHandler.copyChannelData() with dstLength greater than the value field of the available TLV. DstBuffer.length = 6 DstOffset = 0 DstLength = 10	a OUT_OF_TLV_BOUNDARIES ToolkitException is thrown.	
6	CopyChannelData() without Channel Data TLV element 1- call init() method for the RECEIVE DATA proactive command. Call send() method. 2- call ProactiveResponseHandler.copyChannelData() DstBuffer.length = 10 DstOffset = 0		1- RECEIVE DATA proactive command is fetched TERMINAL RESPONSE without ChannelData TLV element.

244

	•	
DstLength = 10		
DSCHEIIGCH - IV		
		l ·

7	Cusassaful sanyChannalData/\	1	A DECENTEDATA
7	Successful copyChannelData()		1- RECEIVE DATA
	Call init() method for the RECEIVE DATA		proactive command is
	proactive command.	3- the Channel Data TLV is copied	fetched
	-	into dstBuffer.	
	Call send() method.		TERMINAL RESPONSE
	2 Call findmin/) with mac of DENICE	The applet checks the returned	with one Channel data TLV
	2- Call findTLV() with TAG of DEVICE IDENTITY.	value is dstOffset + dstLength = 6.	element. (6 bytes available
	IDENIII.	value le deterior i deterigin – e.	= 'Hello2')
			= Tielloz)
	3- Call		
	ProactiveResponseHandler.copyChannelData()		
	DstBuffer.length = 6		
	DstOffset = 0		
	DstLength = 6		
	DstBuffer is the whole Buffer.		
8	Compare copied Buffer		
"	Compare copied Barrer	The applet checks that dstBuffer	
	Check dstBuffer.	contains the channel data from the	
	oneon apopulior.		
		TERMINAL RESPONSE.	
9		The returned byte is the same than	
	Check the Channel Data TLV is selected	the first byte of the Channel data	
		TLV (i.e. 'H')	
		, ,	
	Call the ViewHandler.getValueByte(0)		
	method		
	-		
10	Successful copyChannelData()		
'	oucocciui copy chamicisata()	The Channel Data TLV is copied	
	Call	into dstBuffer.	
	ProactiveResponseHandler.copyChannelData()		
	DstBuffer.length = 6	The applet checks the returned	
	DstOffset = 2	value is dstOffset + dstLength = 5.	
	DstLength = 3		
	DstBuffer is a part of Buffer.		
11	Compare copied Buffer		
	·	The applet checks that bytes from 2	
	Check dstBuffer.	to 4 of dstBuffer contain the first 3	
		bytes of channel data TLV from the	
		TERMINAL RESPONSE.	
12	Successful copyChannelData()	TERMINAL REGI GROE.	
'2	Successful copyonalineibata()	2. The Channel Date TLV is copied	
	1- Initialise dstBuffer to [00, 01]	2- The Channel Data TLV is copied	
	I initialise ustbuller to [00, 01]	into dstBuffer.	
	2- Call		
	ProactiveResponseHandler.copyChannelData()	The returned value is dstOffset +	
	DstBuffer.length = 6	dstLength = 5.	
	DstOffset = 2		
	DstLength = 3		
	DstBuffer is a part of buffer.		i l
13			
10	Compare copied Buffer		
		The applet checks that only bytes	
		The applet checks that only bytes from 2 to 4 of dstBuffer have been	
	Compare copied Buffer	from 2 to 4 of dstBuffer have been	
	Compare copied Buffer	from 2 to 4 of dstBuffer have been updated with the first 3 bytes of	
	Compare copied Buffer	from 2 to 4 of dstBuffer have been updated with the first 3 bytes of channel data TLV from the	
	Compare copied Buffer Check dstBuffer.	from 2 to 4 of dstBuffer have been updated with the first 3 bytes of	4 DECENT DATA
14	Compare copied Buffer	from 2 to 4 of dstBuffer have been updated with the first 3 bytes of channel data TLV from the	1- RECEIVE DATA
	Compare copied Buffer Check dstBuffer. Successful copyChannelData(), with 2 TLV	from 2 to 4 of dstBuffer have been updated with the first 3 bytes of channel data TLV from the TERMINAL RESPONSE.	proactive command is
	Compare copied Buffer Check dstBuffer. Successful copyChannelData(), with 2 TLV 1- call init() method for the RECEIVE DATA	from 2 to 4 of dstBuffer have been updated with the first 3 bytes of channel data TLV from the TERMINAL RESPONSE. 2- the first Channel Data TLV is	
	Compare copied Buffer Check dstBuffer. Successful copyChannelData(), with 2 TLV 1- call init() method for the RECEIVE DATA proactive command.	from 2 to 4 of dstBuffer have been updated with the first 3 bytes of channel data TLV from the TERMINAL RESPONSE.	proactive command is
	Compare copied Buffer Check dstBuffer. Successful copyChannelData(), with 2 TLV 1- call init() method for the RECEIVE DATA	from 2 to 4 of dstBuffer have been updated with the first 3 bytes of channel data TLV from the TERMINAL RESPONSE. 2- the first Channel Data TLV is	proactive command is
	Compare copied Buffer Check dstBuffer. Successful copyChannelData(), with 2 TLV 1- call init() method for the RECEIVE DATA proactive command. Call send() method.	from 2 to 4 of dstBuffer have been updated with the first 3 bytes of channel data TLV from the TERMINAL RESPONSE. 2- the first Channel Data TLV is copied into dstBuffer. The returned value is	proactive command is fetched
	Compare copied Buffer Check dstBuffer. Successful copyChannelData(), with 2 TLV 1- call init() method for the RECEIVE DATA proactive command. Call send() method. 2- call	from 2 to 4 of dstBuffer have been updated with the first 3 bytes of channel data TLV from the TERMINAL RESPONSE. 2- the first Channel Data TLV is copied into dstBuffer.	proactive command is fetched TERMINAL RESPONSE with two Channel data TLV
	Compare copied Buffer Check dstBuffer. Successful copyChannelData(), with 2 TLV 1- call init() method for the RECEIVE DATA proactive command. Call send() method. 2- call ProactiveResponseHandler.copyChannelData()	from 2 to 4 of dstBuffer have been updated with the first 3 bytes of channel data TLV from the TERMINAL RESPONSE. 2- the first Channel Data TLV is copied into dstBuffer. The returned value is	proactive command is fetched TERMINAL RESPONSE with two Channel data TLV element
	Compare copied Buffer Check dstBuffer. Successful copyChannelData(), with 2 TLV 1- call init() method for the RECEIVE DATA proactive command. Call send() method. 2- call ProactiveResponseHandler.copyChannelData() with dstLength lower than the value field	from 2 to 4 of dstBuffer have been updated with the first 3 bytes of channel data TLV from the TERMINAL RESPONSE. 2- the first Channel Data TLV is copied into dstBuffer. The returned value is	proactive command is fetched TERMINAL RESPONSE with two Channel data TLV element 1st TLV: 6 bytes available =
	Compare copied Buffer Check dstBuffer. Successful copyChannelData(), with 2 TLV 1- call init() method for the RECEIVE DATA proactive command. Call send() method. 2- call ProactiveResponseHandler.copyChannelData() with dstLength lower than the value field of the available TLV.	from 2 to 4 of dstBuffer have been updated with the first 3 bytes of channel data TLV from the TERMINAL RESPONSE. 2- the first Channel Data TLV is copied into dstBuffer. The returned value is	proactive command is fetched TERMINAL RESPONSE with two Channel data TLV element 1st TLV: 6 bytes available = 'Hello3'
	Compare copied Buffer Check dstBuffer. Successful copyChannelData(), with 2 TLV 1- call init() method for the RECEIVE DATA proactive command. Call send() method. 2- call ProactiveResponseHandler.copyChannelData() with dstLength lower than the value field of the available TLV. DstBuffer.length = 6	from 2 to 4 of dstBuffer have been updated with the first 3 bytes of channel data TLV from the TERMINAL RESPONSE. 2- the first Channel Data TLV is copied into dstBuffer. The returned value is	proactive command is fetched TERMINAL RESPONSE with two Channel data TLV element 1st TLV: 6 bytes available = 'Hello3' 2nd TLV: 6 bytes available =
	Compare copied Buffer Check dstBuffer. Successful copyChannelData(), with 2 TLV 1- call init() method for the RECEIVE DATA proactive command. Call send() method. 2- call ProactiveResponseHandler.copyChannelData() with dstLength lower than the value field of the available TLV. DstBuffer.length = 6 DstOffset = 0	from 2 to 4 of dstBuffer have been updated with the first 3 bytes of channel data TLV from the TERMINAL RESPONSE. 2- the first Channel Data TLV is copied into dstBuffer. The returned value is	proactive command is fetched TERMINAL RESPONSE with two Channel data TLV element 1st TLV: 6 bytes available = 'Hello3'
14	Compare copied Buffer Check dstBuffer. Successful copyChannelData(), with 2 TLV 1- call init() method for the RECEIVE DATA proactive command. Call send() method. 2- call ProactiveResponseHandler.copyChannelData() with dstLength lower than the value field of the available TLV. DstBuffer.length = 6 DstOffset = 0 DstLength = 6	from 2 to 4 of dstBuffer have been updated with the first 3 bytes of channel data TLV from the TERMINAL RESPONSE. 2- the first Channel Data TLV is copied into dstBuffer. The returned value is	proactive command is fetched TERMINAL RESPONSE with two Channel data TLV element 1st TLV: 6 bytes available = 'Hello3' 2nd TLV: 6 bytes available =
	Compare copied Buffer Check dstBuffer. Successful copyChannelData(), with 2 TLV 1- call init() method for the RECEIVE DATA proactive command. Call send() method. 2- call ProactiveResponseHandler.copyChannelData() with dstLength lower than the value field of the available TLV. DstBuffer.length = 6 DstOffset = 0	from 2 to 4 of dstBuffer have been updated with the first 3 bytes of channel data TLV from the TERMINAL RESPONSE. 2- the first Channel Data TLV is copied into dstBuffer. The returned value is	proactive command is fetched TERMINAL RESPONSE with two Channel data TLV element 1st TLV: 6 bytes available = 'Hello3' 2nd TLV: 6 bytes available =

Check dstBuffer.	first Channel Data TLV from the	
	TERMINAL RESPONSE.	

6.2.8.22.4 Test Coverage

CRR number	Test case number	
N1	7, 10, 12, 14	
N2	14	
N3	9	
N4	8, 11, 13, 15	
P1	1	
P2	2, 3	
P3	4	
P4	5	
C1	6	

6.2.9 Class ToolkitRegistry

6.2.9.1 Method allocateTimer

Test Area Reference: API_2_TKR_ATIM

6.2.9.1.1 Conformance requirement:

The method with following header shall be compliant to its definition in the API.

public byte allocateTimer() throws ToolkitException

6.2.9.1.1.1 Normal execution

- CRRN1: the returned timer identifier shall be between 01 and 08 inclusive.
- CRRN2: the returned timer identifier shall be different from a previously allocated but not released one.
- CRRN3: The SIM Toolkit Framework shall trigger the applet when receiving an ENVELOPE(TIMER EXPIRATION) command for the allocated timer.
- CRRN4: A call to isEventSet() method for EVENT_TIMER_EXPIRATION should return true if the applet has at least one timer allocated.

6.2.9.1.1.2 Parameters error

No requirements.

6.2.9.1.1.3 Context errors

- CRRC1: Shall throw a ToolkitException with reason NO_TIMER_AVAILABLE if all the timers are allocated.
- CRRC2: Shall throw a ToolkitException with reason NO_TIMER_AVAILABLE if the maximum number of timers have been allocated to this applet according to installation parameter.

6.2.9.1.2 Test suite files

Test Script: API_2_TKR_ATIM_1.scr

Test Applet: API_2_TKR_ATIM_1.java

API_2_TKR_ATIM_2.java

API_2_TKR_ATIM_3.java

• Installation parameters:

• For this test procedure the non-volatile memory of each instance is 200 (Hexa).

• The maximum timer parameter value is as follows for each applet:

- applet 1 (API_2_TKR_ATIM_1): 8 timers

- applet 2 (API_2_TKR_ATIM_2): 4 timers

- applet 3 (API_2_TKR_ATIM_3): 0 timer

Load Script: API_2_TKR_ATIM_1.ldr

• The load script installs the 6 instances.

Cleanup Script: API_2_TKR_ATIM_1.clr
Parameter File: API_2_TKR_ATIM_1.par

6.2.9.1.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Allocates up to 8 timers	No exception shall be thrown.	
	(applet 1)	Timer ID returned shall be between	
	8 * allocateTimer().	01 and 08 inclusive. It shall be different after each call.	
2	Allocate timers more than the maximum	Shall throw a ToolkitException with	
	(applet 1)	reason NO_TIMER_AVAILABLE.	
	The applet 1 allocates 1 more timer.		
3	Check applet is Triggered by ENVELOPE(TIMER_EXPIRATION) command (applet1) Send ENVELOPE(TIMER EXPIRATION) with all timers id (not in an increase order).	Shall trigger each time an ENVELOPE(TIMER EXPIRATION) is sent to the SIM, for Timer ID = '01' to '08'.	
	Calls releaseTimer(id) each time a timer expires.		
4	Allocate up to 4 timers		
	<pre>(applet 2) 4 * allocateTimer().</pre>	No exception shall be thrown. Each time, the returned timer identifier shall be between '01' and '08' inclusive. It shall be different after each call.	
5	Allocate timers more than the maximum (applet 3)	Shall throw a ToolkitException with reason NO_TIMER_AVAILABLE.	
	The applet 3 allocates 1 more timer.		

6.2.9.1.4 Test Coverage

CRR number	Test case number	
N1	1, 4	
N2	1, 4	
N3	3	
N4	1	
C1	2	
C2	5	

6.2.9.2 Method changeMenuEntry

Test Area Reference: API_2_TKR_CMETB_BSSBZBS

6.2.9.2.1 Conformance requirement:

The method with following header shall be compliant to its definition in the API.

6.2.9.2.1.1 Normal execution

- CRRN1: The SIM Toolkit Framework shall dynamically update the menu stored in the ME by issuing a SET UP MENU proactive command. The later will reflect the changes done for the entry. The SIM Toolkit Framework shall use the data of the EF sume file in order to build the SET UP MENU command.
- CRRN2: The default state of the changed menu entry is 'enabled'.
- CRRN3: a call to isEventSet() method on EVENT_MENU_SELECTION shall return true before and after the
 call.
- CRRN4: if helpSupported was true then a call to isEventSet() method on EVENT_MENU_SELECTION_HELP_REQUEST event shall return true.
- CRRN5: if helpSupported was true then after the completion of the SETUP MENU command, if an ENVELOPE(MENU_SELECTION_HELP_REQUEST) command is received by the SIM for this entry, then the SIM Toolkit framework shall trigger the applet.
- CRRN6: if help supported was true, the SIM Toolkit Framework shall issue a SETUP MENU command with command qualifier = '80'
- CRRN7: if helpSupported was false and if no entries is supporting help then a call to isEventSet() method on EVENT_MENU_SELECTION_HELP_REQUEST event shall return false.
- CRRN8: if helpSupported was false and if no entries is supporting help then after the completion of the SETUP MENU command, if an ENVELOPE(MENU_SELECTION_HELP_REQUEST) command is received by the SIM, then the SIM Toolkit framework shall not trigger the applet.
- CRRN9: The SIM Toolkit Framework shall supply in the SET UP MENU command with the icon identifier provided in the icon identifier list within the item icon identifier list Simple TLV if all the applets registered to the EVENT_MENU_SELECTION provide it.
- CRRN10: The SIM Toolkit Framework shall set in the SET UP MENU command with the Icon list qualifier transmitted to the ME as 'icon is not self explanatory' if one of the applet registered prefers this qualifier.
- CRRN11: If Next Action Indicator was different from '00', the SIM Toolkit Framework shall issue a SETUP MENU proactive command containing an Items Next Action Indicator simple TLV with the comprehension flag set to 0 as defined in 3GPP TS 51.014 [4].

6.2.9.2.1.2 Parameters error

- CRRP1: Shall throw java.lang.NullPointerException if menuEntry is null
- CRRP2: Shall throw java.lang.ArrayIndexOutOfBoundsException if offset would cause access outside array bounds
- CRRP3: Shall throw java.lang.ArrayIndexOutOfBoundsException if length would cause access outside array bounds

• CRRP4: Shall throw java.lang.ArrayIndexOutOfBoundsException - if both offset and length would cause access outside array bounds

6.2.9.2.1.3 Context errors

- CRRC1: Shall throw a ToolkitException with MENU_ENTRY_NOT_FOUND reason if the Menu Identifier isn't associated to the calling applet instance.
- CRRC2: Shall throw ALLOWED_LENGTH_EXCEEDED if the menu entry string is bigger than the allocated space.

6.2.9.2.2 Test suite files

Additional requirements for the GSM personalization:

- content of EF sume shall be:
 - Title Alpha Identifier: "TOOLKIT TEST"

Test Script: API_2_TKR_CMETB_BBSSBZBS_1.scr

Test Applet: API_2_TKR_CMETB_BBSSBZBS_1.java

- entry '01' is "Init1"
- entry '02' is "Init2"
- Installation parameter:

Same as default applet but with:

- Maximum text length for a menu entry: 15
- Maximum number of menu entries: 2
- Position / Identifier for each menu entry: '01'/'01','02'/'02'

Load Script: API_2_TKR_CMETB_BBSSBZBS_1.ldr

Cleanup Script: API_2_TKR_CMETB_BBSSBZBS_1.clr

Parameter File: API_2_TKR_CMETB_BBSSBZBS_1.par

6.2.9.2.3 Test procedure

ld	Description		API Expectation	APDU Expectation
1	Applet changes the entry's title by menuEntry buffer, with a greater length than the initial length			
	1- ChangeMenuEntry()with parameters:			
	<pre>Id = '02' MenuEntry = "UseAllBuffer" Offset = 0 Length = menuEntry.length NextAction = 0 HelpSupported = false IconQualifier = 0 IconIdentifier = 0.</pre>	2-	No exception shall be thrown. shall return true. shall return false.	
	<pre>2- isEventSet(EVENT_MENU_SELECTION). 3- isEventSet(EVENT_MENU_SELECTION_HELP_R EQUEST).</pre>			The SIM shall issue a SETUP MENU proactive command which contains the new text for entry ID '02'

ld	Description		API Expectation	APDU Expectation
2	Changing the title with part of menuEntry			
	buffer			
	<pre>1- changeMenuEntry()with parameters:</pre>			
	Id = '01'	4	No avagntion about he through	
	MenuEntry = "UsePartOfBuffer"	1-	No exception shall be thrown.	
	Offset = 3	2-	Shall return true.	
	Length = 12 NextAction = 0		Shall return true.	
	HelpSupported = false	3-	Shall return false.	
	IconQualifier = 0	ľ	Onan rotani raioo.	
	<pre>IconIdentifier = 0.</pre>			
	2- isEventSet(EVENT_MENU_SELECTION).			
	z ibavenebee(avani_mano_balaerion).			The SIM shall issue a
	3-			SETUP MENU proactive
	isEventSet(EVENT_MENU_SELECTION_HELP_R			command which contains
	EQUEST)			the new text for entry ID '01'.
3	Length = 0			01.
	<pre>1- changeMenuEntry() for entry '01' and</pre>			
	entry '02', with parameters:			
	Id = '01'/'02'			
	MenuEntry = "LengthEquals0"	4	No avagntion about he through	
	Offset = 0	1-	No exception shall be thrown.	
	Length = 0 NextAction = 0	2-	Shall return true.	
	HelpSupported = false		Shall return true.	
	IconQualifier = 0	3-	shall return false.	
	<pre>IconIdentifier = 0.</pre>			
	2- isEventSet(EVENT_MENU_SELECTION).			
	Z ISEVERESEC(EVENT_MENO_SELECTION).			The CIM of all issues a
	3-			The SIM shall issue a
	isEventSet(EVENT_MENU_SELECTION_HELP_R			SETUP MENU proactive command which contains
	EQUEST).			for entry '01'and entry '02',
				no text part.
4	Setting a next action indicator != 0			
	1- changeMenuEntry()with parameters:			
	i changementaliter () with parameters			
	Id = '02'			
	MenuEntry = "NextActionIndic"			
	Offset = 0 Length = menuEntry.length			
	NextAction = '10' (SETUP CALL)			
	HelpSupported = false			
	IconQualifier = 0			
	<pre>IconIdentifier = 0</pre>			
	2- isEventSet(EVENT_MENU_SELECTION).	1-	No exception shall be thrown.	
			Shall return true.	
	<pre>3- isEventSet(EVENT_MENU_SELECTION_HELP_R</pre>	3-	Shall return false.	
	EQUEST).			
	4- changeMenuEntry()with parameters:			
	TA - 1021			
	<pre>Id = '02' MenuEntry = "NextActionIndic"</pre>			
	Offset = 0			
	Length = menuEntry.length			The SIM shall issue a
	NextAction = '10' (SETUP CALL)			SETUP MENU proactive
	HelpSupported = true IconQualifier = 0			command which contains an
	IconIdentifier = 0			Items Next Action Indicator
				list and which contains a
				command qualifier '80'.

ld	Description	API Expectation	APDU Expectation
5	Checking applet is triggered by a	AiTExpectation	Al Do Expectation
5	MENU_SELECTION_HELP_REQUEST	Applet is trigged by a	
	MICHO_OCCCOTION_TICET _NEGOCOT	MENU_SELECTION_HELP_REQU	
	Send ENVELOPE(MENU_SELECTION_HELP_REQUEST)	EST and the Item Identifier is 02	
	with Item Identifier = '02'		
6	help supported=true		
	1- changeMenuEntry()with parameters:		
	Id = '01'		
	MenuEntry = "HelpSupported"	1- No exception shall be thrown.	
	Offset = 0	The exception chair be timewii.	
	Length = menuEntry.length	2- Shall return true.	
	NextAction = 0	2 Gridii rotarri tradi	
	<pre>HelpSupported = true IconQualifier = 0</pre>	3- Shall return true.	
	IconIdentifier = 0		
	reonracherrier - v		
	2- isEventSet(EVENT_MENU_SELECTION).		
			The SIM shall issue a
	3-		SETUP MENU proactive
	isEventSet(EVENT_MENU_SELECTION_HELP_R		command which contains a
	EQUEST).		command qualifier '80'.
7	Checking applet is triggered by a		de la
'	MENU_SELECTION_HELP_REQUEST	Applet is trigged by a	
	LITO_OLLEO HOR _HELL _INEQUEOR	MENU_SELECTION_HELP_REQU	
	Send ENVELOPE(MENU_SELECTION_HELP_REQUEST)	EST and the Item Identifier is 01	
	with Item Identifier = '01'		
8	Setting icons, help supported = false		
	<pre>1- changeMenuEntry() for entries '01','02', with parameters:</pre>		
	oli, ozi, with parameters.		
	Id = '01'/'02'		
	MenuEntry = "IconQualifier"		
	Offset = 0	1- No exception shall be thrown.	
	Length = menuEntry.length	'	
	NextAction = 0 HelpSupported = false	2- Shall return true.	
	IconQualifier = '01'		
	IconIdentifier = '02' / '01'	Shall return false.	
	2- isEventSet(EVENT_MENU_SELECTION).		
	2_		The SIM shall issue a
	<pre>3- isEventSet(EVENT_MENU_SELECTION_HELP_R</pre>		SETUP MENU proactive
	EQUEST).		command which contains an
			Icon Identifier List.
9	MenuEntry is disabled		
	1- disableMenuEntry('01').		
	2- changeMenuEntry()with parameters:		
	2 Grangerenamery () with parameters.		
	Id = '01'		
	MenuEntry = "EnableEntry"	1- No exception shall be thrown.	
	Offset = 0		
	Length = menuEntry.length NextAction = 0	No exception shall be thrown.	
	NextAction = 0 HelpSupported = false		
	IconQualifier = 0	3- Shall return true.	
	IconIdentifier = 0		
		4- Shall return false.	
	3- isEventSet(EVENT_MENU_SELECTION).		
	4-		The SIM shall issue a
	isEventSet(EVENT_MENU_SELECTION_HELP_R		SETUP MENU proactive
	EQUEST).		command which contains
			the entry. Without Icon
			identifier List Simple TLV

ld	Description	API Expectation	APDU Expectation
10	MenuEntry is null		
	changeMenuEntry()with: MenuEntry = NULL	Shall throw java.lang.NullPointerException.	
11	Offset causes access outside array bounds Id = '01' MenuEntry = "Violation" Offset = menuEntry.length +1 Length = 0 NextAction = 0 HelpSupported = false IconQualifier = 0 IconIdentifier = 0	Shall throw java.lang.ArrayIndexOutOfBoundsE xception.	
	Big Offset causes access outside array bounds Id = '01' MenuEntry = "Violation" Offset = 255 Length = 1 NextAction = 0 HelpSupported = false IconQualifier = 0 IconIdentifier = 0	Shall throw java.lang.ArrayIndexOutOfBoundsE xception.	
13	Offset < 0 causes access outside array bounds Id = '01' MenuEntry = "Violation" Offset = -1 Length = 1 NextAction = 0 HelpSupported = false IconQualifier = 0 IconIdentifier = 0	Shall throw java.lang.ArrayIndexOutOfBoundsE xception.	
14	Length causes access outside array bounds Id = '01' MenuEntry = "Violation" Offset = 0 Length = MenuEntry.length + 1 NextAction = 0 HelpSupported = false IconQualifier = 0 IconIdentifier = 0.	Shall throw java.lang.ArrayIndexOutOfBoundsE xception.	
15	Length < 0 causes access outside array bounds Id = '01' MenuEntry = "Violation" Offset = 0 Length = -1 NextAction = 0 HelpSupported = false IconQualifier = 0 IconIdentifier = 0.	Shall throw java.lang.ArrayIndexOutOfBoundsE xception.	
16	Both offset and length causes access outside array bounds Id = '01' MenuEntry = "Violation" Offset & [1, MenuEntry.length] Length = MenuEntry.length NextAction = 1 HelpSupported = false IconQualifier = 0 IconIdentifier = 0	Shall throw java.lang.ArrayIndexOutOfBoundsException.	

ld	Description	API Expectation	APDU Expectation
17	Invalid ID used		
	<pre>Id = '00' MenuEntry = contains text, != null Offset = 0 Length = menuEntry.length < 16 NextAction = 0 HelpSupported = false IconQualifier = 0 IconIdentifier = 0</pre>	Shall throw a ToolkitException with MENU_ENTRY_NOT_FOUND reason code.	
18	ID isn't allocated to a menu entry of this applet		
	instance		
	<pre>Id = '0A' MenuEntry = contains text, != null Offset = 0 Length = menuEntry.length < 16 NextAction = 0 HelpSupported = false IconQualifier = 0 IconIdentifier = 0</pre>	Shall throw a ToolkitException with reason code: MENU_ENTRY_NOT_FOUND.	
19	The text is bigger than the allocated space		
	<pre>Id = '02' MenuEntry = contains text, != null Offset = 0 Length = menuEntry.length > 15 NextAction = 0 HelpSupported = false IconQualifier = 0 IconIdentifier = 0</pre>	Shall throw a ToolkitException with reason code: ALLOWED_LENGTH_EXCEEDED.	
20	With a smaller text length than the initial length		
	<pre>1. changeMenuEntry()with parameters: Id = '02' MenuEntry = "Init" Offset = 0 Length = menuEntry.length NextAction = 0 HelpSupported = false IconQualifier = 0 IconIdentifier = 0 2. isEventSet(EVENT_MENU_SELECTION) 3. isEventSet(EVENT_MENU_SELECTION_HELD_R</pre>	 No exception shall be thrown. Shall return true. Shall return false. 	The SIM shall issue a SETUP MENU proactive command which contains the new text for entry ID '02'.
	EQUEST)		
	2. isEventSet(EVENT_MENU_SELECTION)		SETUP MENU proactive command which contains the new text for entry ID

6.2.9.2.4 Test Coverage

CRR number	Test case number
N1	1, 2, 3, 4, 6, 8, 9, 20
N2	9
N3	1, 2, 3, 4, 6, 8, 9, 20
N4	6
N5	7,5
N6	6
N7	1, 2, 3, 4, 8, 9, 20
N8	To be checked in framework tests and insert cross reference here
N9	8, 9
N10	8
N11	4
P1	10
P2	11, 12, 13
P3	14, 15
P4	16

CRR number	Test case number	
C1	17, 18	
C2	19	

6.2.9.3 Method clearEvent

Test Area Reference: API_2_TKR_CEVTB

6.2.9.3.1 Conformance requirement:

The method with following header shall be compliant to its definition in the API.

6.2.9.3.1.1 Normal execution

- CRRN1: A call to isEventSet() method for a cleared event should return false after a call to clearEvent.
- CRRN2: The SIM Toolkit Framework shall not trigger the applet on the occurrence of the cleared event anymore.
- CRRN3: if event was EVENT_CALL_CONTROL_BY_SIM and after the call, no applet is registered to it, The SIM Toolkit Framework shall allow an applet to register to this event.
- CRRN4: if event was EVENT_CALL_CONTROL_BY_SIM and one applet is still registered to these event, The SIM Toolkit Framework shall not allow an applet to register to this event.
- CRRN5: if event was EVENT_MO_SHORT_MESSAGE_CONTROL_BY_SIM and after the call, no applet is registered to it, The SIM Toolkit Framework shall allow an applet to register to this event.
- CRRN6: if event was EVENT_MO_SHORT_MESSAGE_CONTROL_BY_SIM and one applet is still registered to these event, The SIM Toolkit Framework shall not allow an applet to set this event.

6.2.9.3.1.2 Parameters error

- CRRP1: Shall throw a Toolkit Exception with reason EVENT_NOT_ALLOWED if event was EVENT_MENU_SELECTION.
- CRRP2: Shall throw a Toolkit Exception with reason EVENT_NOT_ALLOWED if event was EVENT_MENU_SELECTION_HELP_REQUEST.
- CRRP3: Shall throw a Toolkit Exception with reason EVENT_NOT_ALLOWED if event was EVENT_TIMER_EXPIRATION.
- CRRP4: Shall throw a Toolkit Exception with reason EVENT_NOT_ALLOWED if event was EVENT_STATUS_COMMAND.

6.2.9.3.1.3 Context errors

CRRC1: shall throw javacard.framework.TransactionException - if the operation would cause the commit
capacity to be exceeded.

6.2.9.3.2 Test suite files

Test Script: API_2_TKR_CEVTB_1.scr
Test Applet: API_2_TKR_CEVTB_1.java

• As default but applet registers to an event list which contains all defined events in 3GPP TS 43.019 [7] excepted those that are not allowed or supported by setEvent().

Load Script: API_2_TKR_CEVTB_1.ldr

Cleanup script: API_2_TKR_CEVTB_1.clr

Parameter File: API_2_TKR_CEVTB_1.par

6.2.9.3.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Clear ALLOWED unregistered events For events ranging from -1, 1 to 24 and 127* excepted those that aren't allowed	No exception is thrown each time.	
	<pre>(7, 8, 11, 19), the applet calls: 1- clearEvent() method 2- isEventSet() method</pre>	2- Shall return false each time.	
2	Clear registered events		
	1- For each ALLOWED and SUPPORTED event (-1, 1 to 24 and 127)* excepted those that aren't allowed (7, 8, 11, 19), the applet calls setEvent() method. 2- For each ALLOWED and SUPPORTED event (-1, 1 to 24 and 127)* excepted those that aren't allowed (7, 8, 11, 19), the applet calls: 2.1- clearEvent() method 2.2- isEventSet() method	1- No exception shall be thrown.2.1- No exception shall be thrown.	
3	Clearing NOT ALLOWED events For each event among: EVENT_MENU_SELECTION, EVENT_MENU_SELECTION_HELP_REQUEST, EVENT_TIMER_EXPIRATION, EVENT_STATUS_COMMAND 1- The applet calls clearEvent(event) method.	1- Each time, clearEvent shall throw a ToolkitException with reason EVENT_NOT_ALLOWED.	
4	Checking applet isn't triggered by an ENVELOPE(SMS-PP DOWNLOAD) command 1 - reset and initialise the card 2 - An ENVELOPE(SMS-PP DOWNLOAD) is sent with a TAR referencing applet.	Applet is not trigged by an ENVELOPE(SMS-PP DOWNLOAD) command	

NOTE: Although the method clearEvent is defined for a range from -128 to 127 only the allowed events are tested here, because the range from -128 to -2 is reserved for propriatary use in TS 43.019 [7] chapter 6.2 and the range from 25 to 126 is omitted for compatibility with future releases of TS 43.019 [7]

6.2.9.3.4 Test Coverage

CRR number	Test case number
N1	1,2
N2	4
N3	Framework
N4	Framework
N5	Framework
N6	Framework
P1	3
P2	3
P3	3
P4	3

C1	not testable

6.2.9.4 Method disableMenuEntry

Test Area Reference: API_2_TKR_DMETB

6.2.9.4.1 Conformance requirement:

The method with following header shall be compliant to its definition in the API.

6.2.9.4.1.1 Normal execution

- CRRN1: A call to isEventSet() method on EVENT_MENU_SELECTION shall return the same result before and after the call to disableMenuEntry() method.
- CRRN2: A call to isEventSet() method on EVENT_MENU_SELECTION_HELP_REQUEST shall return the same result before and after the call to disableMenuEntry() method.
- CRRN3: After invocation of this method the SIM Toolkit Framework shall dynamically update the menu stored in the ME .
- CRRN4: After invocation of this method, if there is no more enabled menu entries then the SIM Toolkit framework shall issue a SETUP MENU proactive command containing Item Data Object for Item 1 TLV with a length of zero and no value part.

6.2.9.4.1.2 Parameters error

No requirements.

6.2.9.4.1.3 Context errors

• CRRC1: shall throw a ToolkitException with reason = ENTRY_NOT_FOUND if the menu entry doesn't exist for this applet

6.2.9.4.2 Test suite files

Additional requirements for the GSM personalization:

- content of EF sume shall be:
 - Title Alpha Identifier: "TOOLKIT TEST"

Test Script: API_2_TKR_DMETB_1.scr

Test Applet: API_2_TKR_DMETB_1.java

• Installation parameter:

Same as default applet but with:

- Maximum text length for a menu entry: 15

- Maximum number of menu entries: 2

- Position / Identifier for each menu entry: '01'/'01', '02'/'02'

Load Script: API_2_TKR_DMETB_1.ldr

Cleanup script: API_2_TKR_DMETB_1.clr

Parameter File: API_2_TKR_DMETB_1.par

6.2.9.4.3 Test procedure

ld	Description		API Expectation	APDU Expectation
1	Check the menu state before disabling a previously enabled entry not registered to EVENT_MENU_SELECTION_HELP_REQUEST 1- reset and initialise the card 2- isEventSet(EVENT_MENU_SELECTION) 3- isEventSet(EVENT_MENU_SELECTION_HELP_R EQUEST)		Ill return true Ill return false	1- The SIM shall issue a SET UP MENU proactive command with entry '01' and '02'.
2	Check the menu state after disabling a previously enabled entry not registered to EVENT_MENU_SELECTION_HELP_REQUEST 1- disableMenuEntry('01') 2- isEventSet(EVENT_MENU_SELECTION) 3- isEventSet(EVENT_MENU_SELECTION_HELP_REQUEST)	2- Sha	exception shall be thrown. Ill return true. Ill return false.	3- The SIM shall issue a SET UP MENU proactive command with entry '02' only.
3	Check the menu before disabling a previously enabled entry registered to EVENT_MENU_SELECTION_HELP_REQUEST 1- change Menu Entry '02' to indicate help supported 2- isEventSet(EVENT_MENU_SELECTION) 3- isEventSet(EVENT_MENU_SELECTION_HELP_R EQUEST)		Ill return true Ill return true	3- The SIM shall issue a SET UP MENU proactive command with entry '02', indicating help supported.
4	Check the menu after disabling a previously enabled entry registered to EVENT_MENU_SELECTION_HELP_REQUEST 1- disableMenuEntry('02') 2- isEventSet(EVENT_MENU_SELECTION) 3- isEventSet(EVENT_MENU_SELECTION_HELP_R EQUEST)	2- Sha	exception shall be thrown. Il return true. Il return true.	3- The SIM shall issue a SET UP MENU proactive command with 1st Item TLV with a length of 0.
5	Disabling invalid entries For ID ranging from '00' to 'FF' except '01' and '02', the applet calls disableMenuEntry(ID) method.	MENU_	me a Toolkit Exception with ENTRY_NOT_FOUND code shall be thrown.	

6.2.9.4.4 Test Coverage

CRR number	Test case number
N1	1, 2, 3, 4
N2	1, 2, 3, 4
N3	2,4
N4	4
C1	5

6.2.9.5 Method enableMenuEntry

Test Area Reference: API_2_TKR_EMETB

6.2.9.5.1 Conformance requirement:

The method with following header shall be compliant to its definition in the API.

6.2.9.5.1.1 Normal execution

- CRRN1: A call to isEventSet() method on EVENT_MENU_SELECTION shall return the same result before and after the call to enableMenuEntry() method.
- CRRN2: A call to isEventSet() method on EVENT_MENU_SELECTION_HELP_REQUEST shall return the same result before and after the call to enableMenuEntry() method.
- CRRN3:The SIM Toolkit Framework shall dynamically issue a SETUP MENU proactive command which does contain an ITEM SIMPLE TLV object for this entry.

6.2.9.5.1.2 Parameters error

No requirements.

6.2.9.5.1.3 Context errors

• CRRC1: shall throw a ToolkitException with reason = MENU_ENTRY_NOT_FOUND if the menu entry doesn't exist for this applet

6.2.9.5.2 Test suite files

Additional requirements for the GSM personalization:

- content of EF sume shall be:
 - Title Alpha Identifier: "TOOLKIT TEST"

Test Script: API_2_TKR_EMETB_1.scr

Test Applet: API_2_TKR_EMETB_1.java

• Installation parameter:

Same as default applet but with:

- Maximum text length for a menu entry: 15

- Maximum number of menu entries: 2

- Position / Identifier for each menu entry: '01'/'01', '02'/'02'

Load Script: API_2_TKR_EMETB_1.ldr

Cleanup script: API_2_TKR_EMETB_1.clr

Parameter File: API_2_TKR_EMETB_1.par

6.2.9.5.3 Test procedure

ld	Description		API Expectation	APDU Expectation
1	Check menu state before enabling a previously disabled entry not registered to EVENT_MENU_SELECTION_HELP_REQUEST 1- isEventSet(EVENT_MENU_SELECTION) 2- isEventSet(EVENT_MENU_SELECTION_HELP_R EQUEST) 3- disableMenuEntry('01')	2-	Shall return true Shall return false No exception shall be thrown.	3- The SIM shall issue a SET UP MENU proactive command with entry '02' only.
2	Check menu state after enabling a previously disabled entry not registered to EVENT_MENU_SELECTION_HELP_REQUEST 1- enableMenuEntry('01') 2- isEventSet(EVENT_MENU_SELECTION) 3- isEventSet(EVENT_MENU_SELECTION_HELP_R EQUEST)	2-	No exception shall be thrown. Shall return true. Shall return false.	3- The SIM shall issue a SET UP MENU proactive command with entry '01' and '02'.
3	Check menu state before enabling a previously enabled entry registered to EVENT_MENU_SELECTION_HELP_REQUEST 1- change Menu Entry '02' to indicate help supported 2- isEventSet(EVENT_MENU_SELECTION) 3- isEventSet(EVENT_MENU_SELECTION_HELP_R EQUEST) 4- disableMenuEntry('02')	3-	Shall return true Shall return true No exception shall be thrown	4- The SIM shall issue a SET UP MENU proactive command with entry '01'. The help information available flag.is not verified
4	Check menu state after enabling a previously enabled entry registered to EVENT_MENU_SELECTION_HELP_REQUEST 1- enableMenuEntry('02'). 2- isEventSet(EVENT_MENU_SELECTION) 3- isEventSet(EVENT_MENU_SELECTION_HELP_R EQUEST)	2-	No exception shall be thrown. Shall return true. Shall return true.	3- The SIM shall issue a SET UP MENU proactive command with entries '01' and '02' indicating help supported.
5	Enabling invalid entries For ID ranging from '00' to 'FF' except '01' and '02', the applet calls enableMenuEntry(ID) method.	ME	ich time a Toolkit Exception with ENU_ENTRY_NOT_FOUND ason code shall be thrown.	

6.2.9.5.4 Test Coverage

CRR number	Test case number
N1	1, 2, 3, 4
N2	1, 2, 3, 4
N3	1, 2, 3, 4
C1	5

6.2.9.6 Method getEntry

Test Area Reference: API_2_TKR_GETY

6.2.9.6.1 Conformance requirement:

The method with following header shall be compliant to its definition in the API.

6.2.9.6.1.1 Normal execution

- CRRN1: returns a reference to the applet ToolkitRegistry object of the calling applet.
- CRRN2: Each successive call to getEntry() method shall return the same object.

6.2.9.6.1.2 Parameters error

No requirements.

6.2.9.6.1.3 Context errors

No requirements.

6.2.9.6.2 Test suite files

Test Script: API_2_TKR_GETY_1.scr

Test Applet: API_2_TKR_GETY_1.java

Load Script: API_2_TKR_GETY_1.ldr

Cleanup script: API_2_TKR_GETY_1.clr

Parameter File: API_2_TKR_GETY_1.par

6.2.9.6.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Installalation	Returns a not null ToolkitRegistry	
	In the constructor, the applet instance calls the getEntry() method.	instance.	
2	Check it returns the same entry	Returns the same ToolkitRegistry	
	The applet calls the getEntry() method again.	instance as for test case 1.	

6.2.9.6.4 Test Coverage

CRR number	Test case number
N1	1
N2	2

6.2.9.7 Method getPollInterval

Test Area Reference: API_2_TKR_GPOL

6.2.9.7.1 Conformance requirement:

The method with following header shall be compliant to its definition in the API.

public short getPollInterval()

6.2.9.7.1.1 Normal execution

- CRRN1: shall return a value between 1 and 15300 if applet is registered to EVENT_STATUS_COMMAND event.
- CRRN2: shall return POLL_NO_DURATION value (0) if the toolkit applet is not registered to EVENT_STATUS_COMMAND event.

6.2.9.7.1.2 Parameters error

No requirements.

6.2.9.7.1.3 Context errors

No requirements.

6.2.9.7.2 Test suite files

Test Script: API_2_TKR_GPOL_1.scr

Test Applet: API_2_TKR_GPOL_1.java

Load Script: API_2_TKR_GPOL_1.ldr

Cleanup script: API_2_TKR_GPOL_1.clr

Parameter File: API_2_TKR_GPOL_1.par

6.2.9.7.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Applet isn't registered to EVENT_STATUS_COMMAND getPollInterval().	Shall return 0.	
2	Requesting max duration 1- requestPollInterval(15300)	1- No exception shall be thrown.	
	2- Reset and initialise the card 3- getPollInterval()	3- Shall return a value between 1 and 15300.	
3	Requesting System Duration 1- requestPollInterval(POLL_SYSTEM_DURATI ON) 2- Reset and initialise the card 3- getPollInterval().	 No exception shall be thrown. Shall return a value between 1 and 15300. 	
4	Requesting no Duration 1- requestPollInterval(POLL_NO_DURATION) 2- Reset and initialise the card 3- getPollInterval().	1- No exception shall be thrown.3- Shall return 0.	

6.2.9.7.4 Test Coverage

CRR number	Test case number
N1	2, 3
N2	1, 4

6.2.9.8 Method initMenuEntry

Test Area Reference: API_2_TKR_IMET_BSSBZBS

6.2.9.8.1 Conformance requirement:

The method with following header shall be compliant to its definition in the API.

6.2.9.8.1.1 Normal execution

- CRRN1: The SIM Toolkit Framework shall automatically update the menu stored in the ME by issuing a SETUP MENU proactive command. The later will reflect the changes done for the entry. The SIM Toolkit Framework shall use the data of the EF sume file in order to build the SET UP MENU command.
- CRRN2: a call to isEventSet() method on EVENT_MENU_SELECTION shall return true after the 1st successful call (without an exception).
- CRRN3: if helpSupported was true then a following call to isEventSet() method on EVENT_MENU_SELECTION_HELP_REQUEST event shall return true.
- CRRN4: if helpSupported was true then after the completion of the SETUP MENU command, if an ENVELOPE(MENU_SELECTION_HELP_REQUEST) command is received by the SIM for this entry, then the SIM Toolkit framework shall trigger the applet.
- CRRN5: if help supported was true, the SIM Toolkit Framework shall issue a SETUP MENU command with command qualifier = '80'
- CRRN6: if helpSupported was false and there isn't any menu entry supporting help then a call to isEventSet() method on EVENT_MENU_SELECTION_HELP_REQUEST event shall return false.
- CRRN7: The SIM Toolkit Framework shall supply in the SET UP MENU command with the icon identifier provided in the icon identifier list within the item icon identifier list Simple TLV if all the applets registered to the EVENT_MENU_SELECTION provide it.
- CRRN8: The SIM Toolkit Framework shall set in the SET UP MENU command with the Icon list qualifier transmitted to the ME as 'icon is not self explanatory' if one of the applet registered prefers this qualifier.
- CRRN9: If Next Action Indicator was different from '00', the SIM Toolkit Framework shall issue a SETUP MENU proactive command containing an Items Next Action Indicator simple TLV with the comprehension flag set to 0.
- CRRN10: After the completion of the SETUP MENU command, if an ENVELOPE (MENU_SELECTION) command is received by the SIM for this identifier, then the SIM Toolkit framework shall trigger the applet.

6.2.9.8.1.2 Parameters error

• CRRP1: Shall throw java.lang.NullPointerException - if menuEntry is null

- CRRP2: Shall throw java.lang.ArrayIndexOutOfBoundsException if offset would cause access outside array bounds
- CRRP3: Shall throw java.lang.ArrayIndexOutOfBoundsException if length would cause access outside array bounds
- CRRP4:Shall throw java.lang.ArrayIndexOutOfBoundsException if both offset and length would cause access outside array bounds

6.2.9.8.1.3 Context errors

- CRRC1: Shall throw ALLOWED_LENGTH_EXCEEDED if the menu entry string is bigger than the allocated space
- CRRC2: Shall throw REGISTRY_ERROR if the menu entry cannot be initialised (eg no more item data in applet loading parameter)

6.2.9.8.2 Test suite files

Additional requirements for the GSM personalization:

- content of EF sume shall be:
 - Title Alpha Identifier: "TOOLKIT TEST"
 - Test case trigger:
 - 1- Applet instantiation
 - 2- Menu selection
 - 3- Menu selection Help Supported

Test Script: API_2_TKR_IMET_BSSBZBS_1.scr

Test Applet: API_2_TKR_IMET_BSSBZBS_1.java

• Installation parameter:

Same as default applet but with:

- Maximum text length for a menu entry: 15

- Maximum number of menu entries: 6

Position / Identifier for each menu entry: '01'/'01', '02'/'02', '03'/'03', '04'/'04', '05'/'05', and '06'/'06'

Load Script: API_2_TKR_IMET_BSSBZBS_1.ldr

Cleanup script: API_2_TKR_IMET_BSSBZBS_1.clr

Parameter File: API_2_TKR_IMET_BSSBZBS_1.par

6.2.9.8.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	NULL as parameter to menuEntry	Shall throw a	
	MenuEntry = NULL	java.lang.NullPointerException.	
2	Offset > menuEntry.length	Shall throw	
	MenuEntry = "ToolkitTest" Offset = 12 Length = 0	java.lang.ArrayIndexOutOfBoundsException.	

ld	Description	API Expectation	APDU Expectation
3	Offset < 0		•
	MenuEntry = "ToolkitTest" Offset = -1	Shall throw java.lang.ArrayIndexOutOfBoundsException.	
	Length = 11	·	
4	Offset = 255	Shall throw	
	MenuEntry = "ToolkitTest" Offset = 255 Length = 11	java.lang.ArrayIndexOutOfBoundsException.	
5	Length = menuEntry.length+1		
	MenuEntry = "ToolkitTest" Offset = 0 Length = 12	Shall throw java.lang.ArrayIndexOutOfBoundsException.	
6	Length < 0		
	MenuEntry = "ToolkitTest" Offset = 0 Length = -1	Shall throw java.lang.ArrayIndexOutOfBoundsE xception.	
7	Offset + length > menuEntry.length		
	MenuEntry = "ToolkitTest" Offset = 11 Length = 1	Shall throw java.lang.ArrayIndexOutOfBoundsE xception.	
8	MenuEntry.length > size allocated at loading		
	for each menu entry		
	MenuEntry = "ToolkitTest impossible" Offset = 0 Length = 16	ALLOWED_LENGTH_EXCEEDED ToolkitException is thrown.	
9	Successful call,		
	menuEntry is the whole buffer		
	1- initMenuEntry()		
	MenuEntry = "TOOLKIT TEST 1"	1- No exception shall be thrown,	
	Offset = 0 Length = 14	Shall return ID '01'.	
	NextAction = '00'	2- Shall return true.	
	<pre>HelpSupported = false IconQualifier = '00'</pre>	2- Shall return true.	
	IconIdentifier = 0		
	2- isEventSet(EVENT_MENU_SELECTION)		
10	Successful call, menuEntry part of a buffer		
	1- initMenuEntry()		
	MenuEntry = "1234567TOOLKIT TEST 2"		
	Offset = 7	1- No exception shall be	
	Length = 14 NextAction = '00'	thrown,Shall return ID '02'.	
	HelpSupported = false	2- Shall return false.	
	<pre>IconQualifier = '00' IconIdentifier = 0</pre>		
	2-		
	isEventSet(EVENT_MENU_SELECTION_HELP_R		
	EQUEST)		

ld	Description	API Expectation	APDU Expectation
11	Successful call,		
	menuEntry with help supported		
	1- initMenuEntry()		
	MenuEntry = "TOOLKIT TEST 3"		
	Offset = 0		
	Length = 14	1- No exception shall be thrown,	
	NextAction = '00' HelpSupported = true	Shall return ID '03'	
	IconQualifier = '00'	2- Shall return true.	
	IconIdentifier = 0		
	2-		
	isEventSet(EVENT_MENU_SELECTION_HELP_R		
	EQUEST)		
12	Successful call,		
	menuEntry with an Icon		
	•		
	MenuEntry = "TOOLKIT TEST 4"		
	Offset = 0 Length = 14	 No exception shall be thrown. 	
	NextAction = '00'	2- Shall return ID '04'	
	HelpSupported = false		
	<pre>IconQualifier = '01' [icon not self</pre>		
	explanatory]		
13	IconIdentifier = 1 Successful call,		
13	menuEntry with a next action indication		
	mendentry with a next action indication		
	MenuEntry = "TOOLKIT TEST 5"		
	Offset = 0	1- No exception shall be thrown.	
	Length = 14	2- Shall return ID '05'	
	NextAction = '24' [Select Item] HelpSupported = false		
	IconQualifier = '00'		
	IconIdentifier = 0		
14	Successful call,	No exception shall be thrown, Shall	
	length = 0	return ID '06'.	
	<pre>initMenuEntry()</pre>		
	MenuEntry = "ToolkitTest"		
	Offset = 0		
	Length = 0 NextAction = '00'		
	HelpSupported = false		
	<pre>IconQualifier = '00'</pre>		
	<pre>IconIdentifier = 0</pre>		
4.5	Initialiae mana antonothere allegated at the Pro-		
15	Initialise more entry than allocated at loading		
	MenuEntry = "ToolkitTest"	REGISTRY_ERROR	
	Offset = 0	ToolkitException is thrown.	
	Length = 11		

ld	Description	API Expectation	APDU Expectation
16	Dynamic update of the menu stored by the ME Fetch		Card shall Send a SetUpMenu Proactive command: [CommandQualifier]=help supported [Alphald]="TOOLKIT TEST" [ItemId=1] = "TOOLKIT TEST 1" [ItemId=2] = "TOOLKIT TEST 2" [ItemId=3] = "TOOLKIT TEST 3" [ItemId=4] = "TOOLKIT TEST 4" [ItemId=5] = "TOOLKIT TEST 5" [ItemId=6] = "" [ItemSNextAction]=0600000 0002400
17	Check Applet is triggered by ENVELOPE(MENU_SELECTION) command Menu Entry ID = '01'	Applet is trigged by an ENVELOPE(MENU_SELECTION) command & Menu Entry ID = '01'	
18	Check Applet is triggered by ENVELOPE(MENU_SELECTION) command Menu Entry ID = '02'	Applet is trigged by an ENVELOPE(MENU_SELECTION) command & Menu Entry ID = '02'	
19	Check Applet is triggered by ENVELOPE(MENU_SELECTION) command Menu Entry ID = '03'	Applet is trigged by an ENVELOPE(MENU_SELECTION) command & Menu Entry ID = '03'	
20	Check Applet is triggered by ENVELOPE(MENU_SELECTION) command Menu Entry ID = '04'	Applet is trigged by an ENVELOPE(MENU_SELECTION) command & Menu Entry ID = '04'	
21	Check Applet is triggered by ENVELOPE(MENU_SELECTION) command Menu Entry ID = '05'	Applet is trigged by an ENVELOPE(MENU_SELECTION) command & Menu Entry ID = '05'	
22	Check Applet is triggered by ENVELOPE (MENU_SELECTION_HELP_REQUEST) command		
	Menu Entry ID = '03'	Applet is trigged by an ENVELOPE(MENU_SELECTION_HELP_REQUEST) command & Menu Entry ID = '03'	
23	Check Applet is triggered by ENVELOPE(MENU_SELECTION) command Menu Entry ID = '06'	Applet is trigged by an ENVELOPE(MENU_SELECTION) command & Menu Entry ID = '06'	

6.2.9.8.4 Test Coverage

CRR number	Test case number
N1	16
N2	9
N3	11

CRR number	Test case number
N4	22
N5	11, 16
N6	10
N7	12,16
N8	12,16
N9	13,16
N10	9, 10, 11, 12, 13, 14, 17, 18, 19, 20, 21, 23
P1	1
P2	2, 3, 4
P3	5, 6
P4	7
C1	8
C2	14

6.2.9.9 Method is Event Set

Test Area Reference: API_2_TKR_IEVSB

6.2.9.9.1 Conformance requirement:

The method with following header shall be compliant to its definition in the API.

public boolean isEventSet(byte event)

6.2.9.9.1.1 Normal execution

- CRRN1: shall return true if the event is set in the Toolkit Registry for the applet.
- CRRN2: shall return false if the event isn't set in the Toolkit Registry for the applet.

6.2.9.9.1.2 Parameters error

No requirements.

6.2.9.9.1.3 Context errors

No requirements.

6.2.9.9.2 Test suite files

Test Script: API_2_TKR_IEVSB_1.scr

Test Applet: API_2_TKR_IEVSB_1.java

API_2_TKR_IEVSB_2.java

• Installation parameter:

Same as default applet but with:

- Maximum text length for a menu entry: 15

- Maximum number of menu entries: 1

- Position / Identifier for each menu entry: '01'/'01'

- Maximum number of timers: 1

Load Script: API_2_TKR_IEVSB_1.ldr

Cleanup script: API_2_TKR_IEVSB_1.clr

Parameter File: API_2_TKR_IEVSB_1.par

6.2.9.9.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Install Applet1 only registered to EVENT FORMATTED_SMS_PP_ENV and EVENT_MENU_SELECTION Test that events aren't set Applet calls isEventSet() for each event ranging from -1, 1 to 24 and 127* excepted EVENT_FORMATTED_SMS_PP_ENV (2) and EVENT_MENU_SELECTION (7).	Shall return false each time.	
2	For EVENT_FORMATTED_SMS_PP_ENV isEventSet(EVENT_FORMATTED_SMS_PP_ENV)	Shall return true.	
3	For EVENT_MENU_SELECTION	Shall return true	
4	isEventSet(EVENT_MENU_SELECTION) After clearing	Shair rotarri ti us	
4	EVENT_FORMATTED_SMS_PP_ENV 1- clearEvent(EVENT_FORMATTED_SMS_PP_ENV) 2- isEventSet(EVENT_FORMATTED_SMS_PP_ENV)	1- No exception shall be thrown.2- Shall return false.	
5	Setting events For all allowed events defined in TS 43.019[7] for method setEvent(): EVENT_PROFILE_DOWNLOAD, EVENT_FORMATTED_SMS_PP_ENV, EVENT_FORMATTED_SMS_PP_UPD, EVENT_FORMATTED_SMS_PP_UPD, EVENT_UNFORMATTED_SMS_PP_ENV, EVENT_UNFORMATTED_SMS_PP_UPD, EVENT_UNFORMATTED_SMS_PP_UPD, EVENT_UNFORMATTED_SMS_PP_UPD, EVENT_UNFORMATTED_SMS_CB, EVENT_CALL_CONTROL_BY_SIM, EVENT_EVENT_DOWNLOAD_MT_CALL, EVENT_EVENT_DOWNLOAD_MT_CALL, EVENT_EVENT_DOWNLOAD_CALL_DISCONNECTED, EVENT_EVENT_DOWNLOAD_LOCATION_STATUS, EVENT_EVENT_DOWNLOAD_LOCATION_STATUS, EVENT_EVENT_DOWNLOAD_IDLE_SCREEN_AVAILABLE , EVENT_EVENT_DOWNLOAD_LANGUAGE_SELECTION, EVENT_EVENT_DOWNLOAD_BROWSER_TERMINATION, EVENT_EVENT_DOWNLOAD_DATA_AVAILABLE, EVENT_EVENT_DOWNLOAD_CHANNEL_STATUS, EVENT_EVENT_DOWNLOAD_CHANNEL_STATUS, EVENT_EVENT_DOWNLOAD_CHANNEL_STATUS, EVENT_EVENT_DOWNLOAD_CHANNEL_STATUS, EVENT_EVENT_DOWNLOAD_CHANNEL_STATUS, EVENT_FIRST_COMMAND_AFTER_SELECT, EVENT_UNRECOGNIZED_ENVELOPE applet calls: 1- setEvent() method 2- isEventSet() method	1- No exception shall be thrown.2- Shall return true each time.	
6	For EVENT_MENU_SELECTION_HELP_	1- Shall return false.	
	3- isEventSet(EVENT_MENU_SELECTION_HELP_ REQUEST)	Shall return true.	
7	For EVENT_TIMER_EXPIRATION	1- Shall return false.	

	<pre>1- isEventSet(EVENT_TIMER_EXPIRATION) 2- call allocateTimer() 3- isEventSet(EVENT_TIMER_EXPIRATION)</pre>	3- Shall return true.
8	For EVENT_STATUS_COMMAND	
	1- isEventSet(EVENT_STATUS_COMMAND) 2- call requestPollInterval(POLL_SYSTEM_DURATION) 3- isEventSet(EVENT_STATUS_COMMAND)	1- Shall return false. 3- Shall return true.
9	Install Applet2 only registered to	
	EVENT FORMATTED_SMS_PP_ENV	
	isEventSet(EVENT_MENU_SELECTION)	Shall return false.

NOTE: Although the method isEventSet() is defined for a range from –128 to 127 only the allowed events are tested, because the range from -128 to –2 is reserved for propriatary use in TS 43.019 [7] chapter 6.2 and the range from 25 to 126 is omitted for compatibility with future releases of TS 43.019 [7]

6.2.9.9.4 Test Coverage

CRR number	Test case number	
N1	2,3,4,5,6,7,8	
N2	1,5,6,7,8,9	

6.2.9.10 Method releaseTimer

Test Area Reference: API_2_TKR_RTIMB

6.2.9.10.1 Conformance requirement:

The method with following header shall be compliant to its definition in the API.

6.2.9.10.1.1 Normal execution

- CRRN1: if it was the last allocated timer for the applet then a following call to isEventSet() method for EVENT_TIMER_EXPIRATION should return false.
- CRRN2: if applet has timers allocated then a call to isEventSet(EVENT_TIMER_EXPIRATION) shall return true.
- CRRN3: After invocation of the method the indicated timer shall be released and available for reallocation.
- CRRN4: The applet is deregistered of the EVENT_TIMER_EXPIRATION for the indicated Timer Identifier.

6.2.9.10.1.2 Parameters error

 CRRP1: shall throw a ToolkitException with INVALID_TIMER_ID reason if the timer identifier isn't between 1 and 8.

6.2.9.10.1.3 Context errors

• CRRC1: shall throw a ToolkitException with INVALID_TIMER_ID reason if the timer is valid but isn't allocated to this applet.

6.2.9.10.2 Test suite files

Test Script: API_2_TKR_RTIMB_1.scr

Test Applet: API_2_TKR_RTIMB_1.java

• Installation parameter:

- As Default, except max timer which is set to 8.

Load Script: API_2_TKR_RTIMB_1.ldr

Cleanup script: API_2_TKR_RTIMB_1.clr

Parameter File: API_2_TKR_RTIMB_1.par

6.2.9.10.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Releasing not allocated timers For each timer ID ranging from '00' to 'FF', applet calls releaseTimer(ID).	Each time, method shall throw a ToolkitException with reason code INVALID_TIMER_ID.	
2	Releasing allocated timers		
	1- 8 * allocateTimer() .	1- No exception shall be thrown.2- Each time, no exception shall be	
	2- 7 * releaseTimer(id).	thrown.	
	3- isEventSet(EVENT_TIMER_EXPIRATION)	3- Shall return true	
3	Releasing invalid timer ID	1- Shall throw a ToolkitException	
	1- releaseTimer('FF') method	with INVALID_TIMER_ID reason code.	
	2- isEventSet(EVENT_TIMER_EXPIRATION)	2- Shall return true.	
4	Releasing last timer		
	1- releaseTimer(last timer allocated)	 No exception shall be thrown. 	
	2- isEventSet(EVENT_TIMER_EXPIRATION)	2- Shall return false.	
5	Checking we can allocate timers after they have been released		
	8 * allocateTimer().	No exception shall be thrown.	
6	Releasing all timers.		
	For 1 to 8, releaseTimer(id).	No exception shall be thrown.	
7	Checking applet isn't triggered by ENVELOPE(TIMER_EXPIRATION) command Send ENVELOPE(TIMER_EXPIRATION)	Applet is not trigged by an ENVELOPE(TIMER_EXPIRATION) command	

6.2.9.10.4 Test Coverage

CRR number	Test case number	
N1	4	
N2	2, 3	
N3	5, 6	
N4	7	
P1	1, 3	
C1	Framework	

6.2.9.11 Method requestPollInterval

Test Area Reference: API_2_TKR_RPOLS

6.2.9.11.1 Conformance requirement:

The method with following header shall be compliant to its definition in the API.

6.2.9.11.1.1 Normal execution

- CRRN1: If duration is between 1 and 15300 or equal to POLL_SYSTEM_DURATION, the applet registers to EVENT_STATUS_COMMAND.
- CRRN2: If duration is POLL_NO_DURATION, the applet is deregistered from EVENT_STATUS_COMMAND.

6.2.9.11.1.2 Parameters error

• CRRP1: the method should throw a ToolkitException with REGISTRY_ERROR reason if duration is > 15300 or is < -1 (POLL_SYSTEM_DURATION).

6.2.9.11.1.3 Context errors

No requirements.

6.2.9.11.2 Test suite files

Test Script: API_2_TKR_RPOLS_1.scr

Test Applet: API_2_TKR_RPOLS_1.java

Load Script: API_2_TKR_RPOLS_1.ldr

Cleanup script: API_2_TKR_RPOLS_1.clr

Parameter File: API_2_TKR_RPOLS_1.par

6.2.9.11.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Requesting a value between 1 and 15300 s	·	
	1- isEventSet(EVENT_STATUS_COMMMAND)	1- Shall return false.	
	2- requestPollInterval(duration) for boundaries values: 1, 255, 256, 15300.	2- No exception shall be thrown.	
	3- isEventSet(EVENT_STATUS_COMMAND).	3- Shall return true.	
2	Check Applet is triggered by a STATUS command		
	1- reset and card initialization	2- Applet is trigged by a STATUS command	
	2- Send STATUS command		
3	Requesting POLL SYSTEM DURATION		
	1- isEventSet(EVENT_STATUS_COMMMAND).	1- Shall return true.	
	2- RequestPollInterval(POLL_SYSTEM_DURATI	2- No exception shall be thrown.	
	ON).	3- Shall return true.	
	3- IsEventSet(EVENT_STATUS_COMMAND).		

4	Check Applet is triggered by a STATUS command 1- reset and card initialization 2- Send STATUS command	2- Applet is trigged by a STATUS command
5	Requesting invalid duration requestPollInterval(duration) for following values:	Each time, a ToolkitException with REGISTRY_ERROR reason code,
	15301, 32767, -2, -32768	shall be thrown.
6	Requesting POLL NO DURATION	
	1- isEventSet(EVENT_STATUS_COMMMAND)	1- Shall return true.
	2- requestPollInterval(POLL_NO_DURATION)	2- No exception shall be thrown.
	3- isEventSet(EVENT_STATUS_COMMAND)	3- Shall return false.
7	Check Applet isn't triggered by an STATUS	
	command. 1- reset and card initialization 2- Send STATUS command	2- Applet is not trigged by a STATUS command

6.2.9.11.4 Test Coverage

CRR number	Test case number
N1	1, 2, 3, 4
N2	6, 7
P1	5

6.2.9.12 Method setEvent

Test Area Reference: API_2_TKR_SEVTB

6.2.9.12.1 Conformance Requirement:

The method with following header shall be compliant to its definition in the API.

6.2.9.12.1.1 Normal execution

- CRRN1: a following call to isEventSet() method with the same event id shall answer true for the applet.
- CRRN2: the SIM Toolkit Framework shall trigger the applet if an occurrence of the set event happens.
- CRRN3: the method shall accept all the events defined in 3GPP TS 43.019 [7] except: EVENT_MENU_SELECTION, EVENT_MENU_SELECTION_HELP_REQUEST, EVENT_TIMER_EXPIRATION, EVENT_STATUS_COMMAND
- CRRN4: no exception shall be thrown if the applet registers more than once to the same event.
- CRRN5: all updates in the ToolkitRegistry are atomic.

6.2.9.12.1.2 Parameters error

- CRRP1: shall throw a ToolkitException with EVENT_NOT_SUPPORTED reason if event is 0.
- CRRP2: shall throw a ToolkitException with EVENT_NOT_ALLOWED reason if event is EVENT_MENU_SELECTION.

- CRRP3: shall throw a ToolkitException with EVENT_NOT_ALLOWED reason if event is EVENT_MENU_SELECTION_HELP_REQUEST.
- CRRP4: shall throw a ToolkitException with EVENT_NOT_ALLOWED reason if event is EVENT_TIMER_EXPIRATION.
- CRRP5: shall throw a ToolkitException with EVENT_NOT_ALLOWED reason if event is EVENT_STATUS_COMMAND.

6.2.9.12.1.3 Context errors

- CRRC1: shall throw a ToolkitException with EVENT_ALREADY_REGISTERED if event is EVENT_CALL_CONTROL_BY_SIM but another applet is already registered to it.
- CRRC2: shall throw a ToolkitException with EVENT_ALREADY_REGISTERED if event is EVENT_MO_SHORT_MESSAGE_CONTROL_BY_SIM but another applet is already registered to it.
- CRRC3: shall throw a ToolkitException with TAR_NOT_DEFINED if event is FORMATTED_SMS_PP_ENV and the applet has no TAR defined.
- CRRC4: shall throw a ToolkitException with TAR_NOT_DEFINED if event is FORMATTED_SMS_PP_UPD and the applet has no TAR defined.
- CRRC5: shall throw a ToolkitException with TAR_NOT_DEFINED if event is FORMATTED_SMS_CB_ENV and the applet has no TAR defined.
- CRRC6: shall throw javacard.framework.TransactionException if the operation would cause the commit capacity to be exceeded.

6.2.9.12.2 Test suite files

Test Script: API_2_TKR_SEVTB_1.scr

Test Applet: API_2_TKR_SEVTB_1.java

API_2_TKR_SEVTB_2.java

API_2_TKR_SEVTB_3.java

API_2_TKR_SEVTB_4.java

Load Script: API_2_TKR_SEVTB_1.ldr

The load script installs the 4 instances.

Cleanup script: API_2_TKR_SEVTB_1.clr

Parameter File: API_2_TKR_SEVTB_1.par

6.2.9.12.3 Test Procedure

ld	Description	API Expectation	APDU Expectation
1	Applet 1 is triggered by ENVELOPE(SMS_ PP_FORMATTED) command.		
	Send ENVELOPE(SMS_PP_FORMATTED)	Applet 1 shall be triggered	
2	Setting ALLOWED and SUPPORTED events 1- For all allowed events (-1, 1 to 24 and 127 excepted 7, 8, 11, 19) defined in TS 43.019 [7]*: EVENT_PROFILE_DOWNLOAD, EVENT_FORMATTED_SMS_PP_ENV, EVENT_FORMATTED_SMS_PP_UPD, EVENT_FORMATTED_SMS_CB, EVENT_UNFORMATTED_SMS_PP_ENV,		

ld	Description	API Expectation	APDU Expectation
	EVENT_UNFORMATTED_SMS_PP_UPD, EVENT_UNFORMATTED_SMS_CB,		
	EVENT_CALL_CONTROL_BY_SIM,		
	EVENT_MO_SHORT_MESSAGE_CONTROL_BY_SIM,		
	EVENT_EVENT_DOWNLOAD_MT_CALL, EVENT_EVENT_DOWNLOAD_CALL_CONNECTED,		
	EVENT_EVENT_DOWNLOAD_CALL_DISCONNECTED,		
	EVENT_EVENT_DOWNLOAD_LOCATION_STATUS, EVENT_EVENT_DOWNLOAD_USER_ACTIVITY,		
	EVENT_EVENT_DOWNLOAD_IDLE_SCREEN_AVAILABL		
	E,		
	EVENT_EVENT_DOWNLOAD_CARD_READER_STATUS, EVENT_EVENT_DOWNLOAD_LANGUAGE_SELECTION,		
	EVENT_EVENT_DOWNLOAD_BROWSER_TERMINATION,		
	EVENT_EVENT_DOWNLOAD_DATA_AVAILABLE, EVENT_EVENT_DOWNLOAD_CHANNEL_STATUS,		
	EVENT_FIRST_COMMAND_AFTER_SELECT,		
	EVENT_UNRECOGNIZED_ENVELOPE	1.1- No exception shall be thrown.	
	1.1- clearEvent(event)	1.2- Shall return false.	
	1.2- isEventSet(event)	1.3- No exception shall be thrown.	
	1.3- setEvent(event)	1.4- Shall return true.	
	1.4- isEventSet(event)		
	1.5- clearEvent(event)	1.5- No exception shall be thrown.	
3	Event 0		
	Call setEvent(0)	Shall throw a ToolkitException with EVENT_NOT_SUPPORTED reason	
		code.	
4	Setting EVENT_MENU_SELECTION	Chall throw a TaalkitEvaantian with	
	Call setEvent(EVENT_MENU_SELECTION)	Shall throw a ToolkitException with EVENT_NOT_ALLOWED reason	
		code.	
F	Catting		
5	Setting EVENT_MENU_SELECTION_HELP_REQUEST	Shall throw a ToolkitException with	
		EVENT_NOT_ALLOWED reason	
	Call setEvent(EVENT_MENU_SELECTION_HELP_REQUES	code.	
	T)		
6	Setting EVENT_TIMER_EXPIRATION	Shall throw a ToolkitException with	
	Call setEvent(EVENT_TIMER_EXPIRATION)	EVENT_NOT_ALLOWED reason	
		code.	
7	Setting EVENT_STATUS_COMMAND		
′	Jennig Event Status_Command	Shall throw a ToolkitException with	
	Call setEvent(EVENT_STATUS_COMMAND)	EVENT_NOT_ALLOWED reason	
		code.	
8	Setting EVENT_CALL_CONTROL_BY_SIM		
		No Exception shall be thrown	
9	Call setEvent(EVENT_CALL_CONTROL_BY_SIM)		
9	Setting EVENT_MO_SHORT_MESSAGE_CONTROL_B		
	Y_SIM	No Everation -b-U b- 4	
	Call	No Exception shall be thrown	
	setEvent(EVENT_MO_SHORT_MESSAGE_CONTROL_B		
10	Y_SIM) Check applet is triggered by an	Applet is trigged by an	
.0	ENVELOPE(CALL_CONTROL_BY_SIM)	ENVELOPE(CALL_CONTROL_BY_	
	Trigger the applet	SIM)	
11	Check applet is triggered by an	Applet is trigged by an	
	ENVELOPE(MO_SHORT_MESSAGE_CONTRO L_BY_SIM)	E_CONTROL_BY_SIM)	
L			

ld	Description	API Expectation	APDU Expectation
	Trigger the Applet		
12	Applet 2 is triggered by ENVELOPE(SMS_PP_DOWNLOAD) command. Trigger the applet 2	Applet 2 is trigged by an ENVELOPE(SMS_ PP_DOWNLOAD) command	
13	Applet 2 registers to CALL_CONTROL_BY_SIM but it is already assigned setEvent(EVENT_CALL_CONTROL_BY_SIM)	Shall throw a ToolkitException with EVENT_ALREADY_REGISTERED reason code.	
14	Applet 2 registers to MO_MESSAGE_CONTROL_BY SIM but it is already assigned setEvent(EVENT_MO_SHORT_MESSAGE_CONTROL_B Y_SIM)	Shall throw a ToolkitException with EVENT_ALREADY_REGISTERED reason code.	
15	Applet 3 with no TAR defined registers to EVENT_UNFORMATTED_SMS_CB		
	1- send ENVELOPE(CELL_BROADCAST_DATA_ DOWNLOAD)	1- Applet 3 shall be triggered	
	2- setEvent(FORMATTED_SMS_PP_ENV)	2- ToolkitException with reason code TAR_NOT_DEFINED should be thrown	
	3- setEvent(FORMATTED_SMS_PP_UPD)	3- ToolkitException with reason code TAR_NOT_DEFINED should be thrown	
	4- setEvent(FORMATTED_SMS_CB_ENV)	ToolkitException with reason code TAR_NOT_DEFINED should be thrown	
16	Applet 4 registers multiple to EVENT_FORMATTED_SMS_PP_ENV	1- Applet 4 shall be triggered	
	<pre>1- send ENVELOPE(EVENT_FORMATTED_ SMS_PP_ENV)</pre>	2- no Exception shall be thrown	
	2- setEvent(EVENT_FORMATTED_SMS_PP_ UPD)		
	3- setEvent(EVENT_FORMATTED_SMS_PP_ UPD)	3- no Exception shall be thrown	
	4- send ENVELOPE(EVENT_FORMATTED_ SMS_PP_UPD)	4- Applet 4 shall be triggered	

NOTE: Although the method setEvent is defined for a range from -128 to 127 only the allowed events are tested, because the range from -128 to -2 is reserved for propriatary use in TS TS 43.019 [7] chapter 6.2 and the range from 25 to 126 is omitted for compatibility with future releases of TS 43.019 [7]

6.2.9.12.4 Test Coverage

CRR number	Test case number
N1	2
N2	1,8,9,10,11,12
N3	2,4,5,6,7
N4	16
N5	not testable
P1	3
P2	4
P3	5
P4	6
P5	7
C1	13
C2	14
C3	15

C4	15
C5	15
C6	not testable

6.2.9.13 Method setEventList

Test Area Reference: API_2_TKR_SEVL_BSS

6.2.9.13.1 Conformance Requirement:

The method with following header shall be compliant to its definition in the API.

6.2.9.13.1.1 Normal execution

- CRRN1: for all events set successfully by this method, a call to isEventSet() method should return true.
- CRRN2: the SIM Toolkit Framework shall trigger the applet if an occurrence of one of the successfully registered events happens.
- CRRN3: this method shall accept all the events defined in 3GPP TS 43.019 [7] except: EVENT_MENU_SELECTION, EVENT_MENU_SELECTION_HELP_REQUEST, EVENT_TIMER_EXPIRATION, EVENT_STATUS_COMMAND.
- CRRN4: all updates on the ToolkitRegistry are atomic
- CRRN5: No exception shall be thrown if the applet registers more than once to the same event.

6.2.9.13.1.2 Parameters error

- CRRP1: shall throw a java.lang.NullPointerException if eventList is null.
- CRRP2: shall throw a java.lang.ArrayIndexOutOfBoundsException if offset would cause access outside array bounds.
- CRRP3: shall throw a java.lang.ArrayIndexOutOfBoundsException if length would cause access outside array bounds.
- CRRP4: shall throw a java.lang.ArrayIndexOutOfBoundsException if both offset and length would cause access
 outside array bounds.
- CRRP5: shall throw a ToolkitException with EVENT_NOT_SUPPORTED reason if event is 0.
- CRRP6: shall throw a ToolkitException with EVENT_NOT_ALLOWED reason if eventList contains EVENT_MENU_SELECTION.
- CRRP7: shall throw a ToolkitException with EVENT_NOT_ALLOWED reason if eventList contains EVENT_MENU_SELECTION_HELP_REQUEST.
- CRRP8: shall throw a ToolkitException with EVENT_NOT_ALLOWED reason if eventList contains EVENT_TIMER_EXPIRATION.
- CRRP9: shall throw a ToolkitException with EVENT_NOT_ALLOWED reason if eventList contains EVENT STATUS COMMAND.

6.2.9.13.1.3 Context errors

- CRRC1: shall throw a ToolkitException with EVENT_ALREADY_REGISTERED if eventList contains EVENT_CALL_CONTROL_BY_SIM but another applet is already registered to it.
- CRRC2: shall throw a ToolkitException with EVENT_ALREADY_REGISTERED if eventList contains EVENT_MO_SHORT_MESSAGE_CONTROL_BY_SIM but another applet is already registered to it.
- CRRC3: shall throw a ToolkitException with TAR_NOT_DEFINED if event is FORMATTED_SMS_PP_ENV and the applet has no TAR defined.
- CRRC4: shall throw a ToolkitException with TAR_NOT_DEFINED if event is FORMATTED_SMS_PP_UPD and the applet has no TAR defined.
- CRRC5: shall throw a ToolkitException with TAR_NOT_DEFINED if event is FORMATTED_SMS_CB_ENV and the applet has no TAR defined.
- CRRC6: shall throw javacard.framework.TransactionException if the operation would cause the commit capacity to be exceeded.

6.2.9.13.2 Test suite files

Test Script: API_2_TKR_SEVL_BSS_1.scr

Test Applet: API_2_TKR_SEVL_BSS_1.java

API_2_TKR_SEVL_BSS_2.java

API_2_TKR_SEVL_BSS_3.java

Load Script: API_2_TKR_SEVL_BSS_1.ldr

The load script installs the 4 instances.

Cleanup script: API_2_TKR_SEVL_BSS_1.clr

Parameter File: API_2_TKR_SEVL_BSS_1.par

6.2.9.13.3 Test Procedure

ld	Description	API Expectation	APDU Expectation
1	Applet 1 Registering all eventList buffer		
	EventList = all allowed events (-1, 1 to 24 and 127 excepted 7, 8, 11, 19) defined in TS 43.019[7]:		
	EVENT_PROFILE_DOWNLOAD, EVENT_FORMATTED_SMS_PP_ENV, EVENT_FORMATTED_SMS_PP_UPD, EVENT_FORMATTED_SMS_CB, EVENT_UNFORMATTED_SMS_PP_ENV, EVENT_UNFORMATTED_SMS_PP_ENV, EVENT_UNFORMATTED_SMS_PP_UPD, EVENT_UNFORMATTED_SMS_CB, EVENT_CALL_CONTROL_BY_SIM, EVENT_MO_SHORT_MESSAGE_CONTROL_BY_SIM, EVENT_EVENT_DOWNLOAD_MT_CALL, EVENT_EVENT_DOWNLOAD_CALL_CONNECTED, EVENT_EVENT_DOWNLOAD_LOCATION_STATUS, EVENT_EVENT_DOWNLOAD_USER_ACTIVITY, EVENT_EVENT_DOWNLOAD_IDLE_SCREEN_AVAILABLE , EVENT_EVENT_DOWNLOAD_LANGUAGE_SELECTION, EVENT_EVENT_DOWNLOAD_BROWSER_TERMINATION, EVENT_EVENT_DOWNLOAD_DATA_AVAILABLE, EVENT_EVENT_DOWNLOAD_CHANNEL_STATUS, EVENT_EVENT_DOWNLOAD_CHANNEL_STATUS, EVENT_EVENT_DOWNLOAD_CHANNEL_STATUS, EVENT_FIRST_COMMAND_AFTER_SELECT, EVENT_UNRECOGNIZED_ENVELOPE 1- For each event in EventList clearEvent(event)	1- No exception shall be thrown.	
	2- setEventList(eventList)	·	
	Offset = 0 Length = eventList.lentgh	2- No exception shall be thrown.	
	3- For all events in eventList isEventSet(event)4- For each event in EventList clearEvent(event)	3- Each time shall return true.4- No exception shall be thrown.	
2	Registering part of eventList buffer		
	EventList = all allowed events defined in TS 43.019[7] (see test case 1). 1- For each event in EventList	1- No exception shall be thrown.	
	clearEvent(event) 2- setEventList(eventList, offset, length)	2- No exception shall be thrown.	
	Offset > 0 Length = eventList.lentgh - offset	3- Each time shall return true for events ranging from offset to offset+length else shall return false.	
	<pre>3- For all events in eventList: isEventSet(event)</pre>	4- No exception shall be thrown.	
	<pre>4- For each event in EventList: clearEvent(event)</pre>		
3	Null buffer	Shall throw a	
	EventList = null	java.lang.NullPointerException Exception	
4	Out of bounds offset	Shall throw a java.lang.ArrayIndexOutOfBounds	
	Offset = eventList.length Length = 1	Exception	

ld	Description	API Expectation	APDU Expectation
_	Out of hounds and him offers		
5	Out of bounds and big offset Offset = 255 Length = 1	Shall throw a java.lang.ArrayIndexOutOfBounds Exception	
6	Offset < 0	Shall throw a	
	Offset = -1 Length = 1	java.lang.ArrayIndexOutOfBounds Exception	
7	Out of bounds length	G	
	Offset = 0 Length = eventList.length + 1	Shall throw a java.lang.ArrayIndexOutOfBounds Exception	
8	Out of bounds and big length	Chall throws	
	Offset = 0 Length = 255	Shall throw a java.lang.ArrayIndexOutOfBounds Exception	
9	Length < 0	Shall throw a	
	Offset = 0 Length = -1	java.lang.ArrayIndexOutOfBounds Exception	
10	Out of bounds offset + Length	Shall throw a	
	Offset + length > eventList.length + 1	java.lang.ArrayIndexOutOfBounds Exception	
11	Event 0	Chall throw a Tablkit Evantion with	
	Call setEventList(eventList) with eventList indicating event 0	Shall throw a ToolkitException with EVENT_NOT_SUPPORTED reason code.	
12	EVENT_MENU_SELECTION		
	Call setEventList(eventList) with eventList indicating EVENT_MENU_SELECTION	Shall throw a ToolkitException with reason code EVENT_NOT_ALLOWED.	
13	EVENT_MENU_SELECTION_HELP_REQUEST		
	Call setEventList(eventList) with eventList indicating EVENT_MENU_SELECTION_HELP_REQUEST	Shall throw a ToolkitException with reason code EVENT_NOT_ALLOWED.	
14	EVENT_TIMER_EXPIRATION		
	Call setEventList(eventList) with eventList indicating EVENT_TIMER_EXPIRATION	Shall throw a ToolkitException with reason code EVENT_NOT_ALLOWED.	
15	EVENT_STATUS_COMMAND		
	Call setEventList(eventList) with eventList indicating EVENT_STATUS_COMMAND	Shall throw a ToolkitException with reason code EVENT_NOT_ALLOWED.	
16	Setting EVENT_CALL_CONTROL_BY_SIM		
	setEventList(List, 0, 2) with List containing EVENT_CALL_CONTROL_BY_SIM & EVENT_MO_SHORT_MESSAGE_CONTROL_BY_SIM	Shall not throw an exception	
17	Check applet is triggered by an ENVELOPE(CALL_CONTROL_BY_SIM) Reset and initialise the card Trigger the applet	Applet is trigged by an ENVELOPE(CALL_CONTROL_BY _SIM)	
18	Check applet is triggered by an ENVELOPE(MO_SHORT_MESSAGE_CONTROLBY_SIM) Trigger the applet	Applet is trigged by an ENVELOPE(MO_SHORT_MESSA GE_CONTROL_BY_SIM)	
L	55:	I	

ld	Description	API Expectation	APDU Expectation
19	Applet 2 registers to CALL_CONTROL_BY_SIM but it is already assigned setEventList(MonoEventList,0,1) with MonoEventList containing EVENT_CALL_CONTROL_BY_SIM	Shall throw a ToolkitException with EVENT_ALREADY_REGISTERED reason code.	
20	Applet 2 registers to MO_SHORT_MESSAGE_CONTROL_BY_SIM but it is already assigned setEventList(MonoEventList,0,1) with MonoEventList containing EVENT_MO_SHORT_MESSAGE_CONTROL_BY _SIM	Shall throw a ToolkitException with EVENT_ALREADY_REGISTERED reason code.	
21	Applet 3 with no TAR defined registers to EVENT_UNFORMATTED_SMS_CB		
	1- send ENVELOPE(EVENT_UNFORMATTED_SMS_CB) 2-	1- Applet3 shall be triggered	
	<pre>setEventList(EVENT_FORMATTED_SMS_PP_ENV , EVENT_UNFORMATTED_SMS_PP_ENV, EVENT_UNFORMATTED_SMS_PP_ENV)</pre>	2- ToolkitException with reason code TAR_NOT_DEFINED should be thrown	
	3- setEventList(EVENT_UNFORMATTED_SMS_PP_ ENV, EVENT_FORMATTED_SMS_PP_UPD, EVENT_UNFORMATTED_SMS_PP_ENV)	3- ToolkitException with reason code TAR_NOT_DEFINED should be thrown	
	setEventList(EVENT_UNFORMATTED_SMS_PP_E NV, EVENT_UNFORMATTED_SMS_PP_ENV, EVENT_FORMATTED_SMS_CB_ENV) 5-	4- ToolkitException with reason code TAR_NOT_DEFINED should be thrown	
	isEventSet(EVENT_UNFORMATTED_SMS_PP_ENV)	5- method should return FALSE	
	6- isEventSet(EVENT_UNFORMATTED_SMS_PP_UPD)	6- method should return FALSE 7- method should return FALSE	
	7- isEventSet(EVENT_FORMATTED_SMS_PP_ENV)	8- method should return FALSE	
	8- isEventSet(EVENT_FORMATTED_SMS_PP_UPD)	9- method should return FALSE	
-	9- isEventSet(EVENT_FORMATTED_SMS_CB_ENV)		
22	1- setEventList(EVENT_UNFORMATTED_SMS_PP_E NV, EVENT_UNFORMATTED_SMS_PP_ENV)	1- no exception should be thrown	
	2- isEventSet(EVENT_UNFORMATTED_SMS_PP_ENV)	2- method should return true	

6.2.9.13.4 Test Coverage

CRR number	Test case number	
N1	1,2	
N2	16,17,18	
N3	1,2,11,12,13,14,15	
N4	21	
N5	22	
P1	3	
P2	4,5,6	
P3	7,8,9	
P4	10	
P5	11	
P6	12	

P7	13
P8	14
P9	15
C1	19
C2	20
C3	21
C4	21
C5	21
C6	not testable

6.2.10 Class ViewHandler

It is not possible to test the methods provided by this class as it is declared 'abstract'; it will be done in the class inheriting it: EditHandler, EnvelopeHandler, ProactiveResponseHandler, ProactiveHandler.

6.2.11 Class ToolkitException

6.2.11.1 Exception Constants

Test Area Reference: API_2_TKE_CONS

6.2.11.1.1 Conformance requirement:

There is no API, only constants.

6.2.11.1.1.1 Normal execution

• CRRN1: The Constants of the class ToolkitException shall all have the same name and value defined in the 3GPP TS 43.019 [7].

6.2.11.1.1.2 Parameters error

No requirements.

6.2.11.1.1.3 Context errors

No requirements.

6.2.11.1.2 Test suite files

None.

6.2.11.1.3 Test procedure

The constants in Java are resolved at compilation time, therefore a runtime test is not useful. No test of constants will be performed.

6.2.11.2 Constructor ToolkitException

Test Area Reference: API_2_TKE_COORS

6.2.11.2.1 Conformance requirement:

The constructor with following headershall compliant to its definition in the API.

public ToolkitException(short reason)

6.2.11.2.1.1 Normal execution

• CRRN1: Construct a ToolkitException instance with the specified reason.

6.2.11.2.1.2 Parameters error

No requirements.

6.2.11.2.1.3 Context errors

No requirements.

6.2.11.2.2 Test suite files

No additional requirements for the GSM personalization:

Test Script: API_2_TKE_COORS_1.scr

Test Applet: API_2_TKE_COORS_1.java

Load Script: API_2_TKE_COORS_1.ldr

Cleanup script: API_2_TKE_COORS_1.clr

Parameter File: API_2_TKE_COORS_1.par

6.2.11.2.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	reason = (short) 19	ToolkitException.getReason() =	
		(short)19	

6.2.11.2.4 Test Coverage

CRR number	Test case number
N1	1

6.2.11.3 Method throwlt

Test Area Reference: API_2_TKE_THITS

6.2.11.3.1 Conformance requirement:

The method with following header shall compliant to its definition in the API.

6.2.11.3.1.1 Normal execution

- CRRN1: Throws the JCRE instance of the ToolkitException class with the specified reason.
- CRRN2: extends javacard.framework.CardRuntimeException

6.2.11.3.1.2 Parameters error

No requirements.

6.2.11.3.1.3 Context errors

No requirements.

6.2.11.3.2 Test suite files

No additional requirements for the GSM personalization:

Test Script: API_2_TKE_THITS_1.scr

Test Applet: API_2_TKE_THITS_1.java

Load Script: API_2_TKE_THITS_1.ldr

Cleanup Script: API_2_TKE_THITS_1.clr

Parameter File: API_2_TKE_THITS_1.par

6.2.11.3.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Throws the JCRE instance of ToolkitException	Reason = 0	
	with the specified reason		
2	Throws the JCRE instance of ToolkitException	Reason = 1	
	with the specified reason		
3	Throws the JCRE instance of ToolkitException	Reason = 15	
	with the specified reason		
4	ToolkitException extends	Reason = 0	
	javacard.framework.CardRuntimeException		
5	ToolkitException extends	Reason = 1	
	javacard.framework.CardRuntimeException		
6	ToolkitException extends	Reason = 15	
	javacard.framework.CardRuntimeException		

6.2.11.3.4 Test Coverage

CRR number	Test case number
N1	1, 2, 3
N2	4, 5, 6

6.3 SIM Toolkit Framework

6.3.1 Minimum Handler Availability

This test area tests the rules that define the minimum requirements for the availability of the system handlers.

6.3.1.1 ProactiveHandler

Test Area Reference: FWK_MHA_PAHD

6.3.1.1.1 Conformance Requirement

6.3.1.1.1.1 Normal Execution

• CRRN1: If a proactive session is not ongoing the ProactiveHandler is available from the invocation to the termination of the processToolkit method for the following events:

EVENT_FORMATTED_SMS_PP_ENV

EVENT_UNFORMATTED_SMS_PP_ENV

EVENT_FORMATTED_SMS_PP_UPD

EVENT_UNFORMATTED_SMS_PP_UPD

EVENT_FORMATTED_SMS_CB

EVENT_UNFORMATTED_SMS_CB

EVENT_MENU_SELECTION

EVENT_MENU_SELECTION_HELP_REQUEST

EVENT_TIMER_EXPIRATION

EVENT_EVENT_DOWNLOAD_MT_CALL

EVENT_EVENT_DOWNLOAD_CALL_CONNECTED

EVENT_EVENT_DOWNLOAD_CALL_DISCONNECTED

EVENT_EVENT_DOWNLOAD_LOCATION_STATUS

EVENT_EVENT_DOWNLOAD_USER_ACTIVITY

EVENT_EVENT_DOWNLOAD_IDLE_SCREEN_AVAILABLE

EVENT_EVENT_DOWNLOAD_CARD_READER_STATUS

EVENT_EVENT_DOWNLOAD_LANGUAGE_SELECTION

EVENT_EVENT_DOWNLOAD_BROWSER_TERMINATION

EVENT_UNRECOGNIZED_ENVELOPE

EVENT_STATUS_COMMAND

EVENT_CALL_CONTROL

EVENT_SMS_MO_CONTROL

EVENT_PROFILE_DOWNLOAD

EVENT_EVENT_DOWNLOAD_DATA_AVAILABLE

EVENT_EVENT_DOWNLOAD_CHANNEL_STATUS

6.3.1.1.1.2 Parameters error

No requirements.

6.3.1.1.3 Context errors

• CRRC1: The ProactiveHandler and its content are not available for any toolkit applet triggered from the invocation to the termination of their processToolkit method for the following events:

EVENT_FIRST_COMMAND_AFTER_SELECT

6.3.1.1.2 Test Suite Files

Test Script: FWK_MHA_PAHD_1.scr

Test Applet: FWK_MHA_PAHD_1.java

FWK_MHA_PAHD_2.java

Load Script: FWK_MHA_PAHD_1.ldr

Cleanup Script: FWK_MHA_PAHD_1.clr

Parameter File: FWK_MHA_PAHD_1.par

Test Procedure

ld	Description	API /Framework Expectation	APDU Expectation
1	Applets registration to all events and Proactive		
	Handler availability with		
	EVENT_FIRST_COMMAND_AFTER_SELECT		
	Applet1 is registered to all events defined in TS 43.019 [7].		
	Using the methods initMenuEntry() for		
	<pre>EVENT_MENU_SELECTION, requestPollInterval() for</pre>		
	<pre>EVENT_STATUS_COMMAND, allocateTimer() for EVENT_TIMER_EXPIRATION and setEventList()</pre>		
	for the rest of the events.		
	Applet2 is registered to all events		
	defined in TS 43.019 [7],		
	EVENT_CALL_CONTROL_BY_SIM and EVENT_MO_SHORT_MESSAGE_CONTROL_BY_SIM.		
	Using the methods initMenuEntry() for		
	EVENT_MENU_SELECTION, requestPollInterval() for		
	EVENT_STATUS_COMMAND, allocateTimer() for		
	<pre>EVENT_TIMER_EXPIRATION and setEventList() for the rest of the events.</pre>		
	The priority of applet1 is higher than priority of applet2		
	1- Select MF		
	2- Applet1 gets the Proactive Handler.	1- Applet1 is triggered by EVENT_FIRST_COMMAND_AFTE	
	Applet1 is deregistered from EVENT_FIRST_COMMAND_AFTER_SELECT.	R_SELECT	
	EVENI_FIRSI_COMMAND_AFIEK_SEDECT.		
	3- Applet2 gets the Proactive Handler		
	Applet2 is deregistered to	2- A Toolkit Exception	
	EVENT_FIRST_COMMAND_AFTER_SELECT.	HANDLER_NOT_AVAILABLE is	
		thrown.	
		Applet1 finalizes	
		Applet2 is triggered by	
		EVENT_FIRST_COMMAND_AFTE	
		R_SELECT	
		3- A Toolkit Exception HANDLER_NOT_AVAILABLE is	
		thrown.	
		Applet2 finalizes	

ld	Description	API /Framework Expectation	APDU Expectation
2	Proactive Handler availability with	1- Applet1 is triggered by	
	EVENT_PROFILE_DOWNLOAD	EVENT_PROFILE_DOWNLOAD	
	1- Terminal Profile command is sent to the SIM without the facility of		
	SET_EVENT_LIST, POLL_INTERVAL, SET UP IDLE	2- No exception is thrown.	
	MODE TEXT and SET UP MENU.	Applet1 finalizes.	
	2- Applet1 gets the Proactive Handler Applet1 is deregistered to	Applet2 is triggered by	
	EVENT_PROFILE_DOWNLOAD	EVENT_PROFILE_DOWNLOAD	
		3- No exception is thrown	
	3- Applet2 gets the Proactive Handler Applet2 is deregistered to		
	EVENT_PROFILE_DOWNLOAD		
3	Proactive Handler availability with		
	EVENT_MENU_SELECTION_HELP_REQUEST		
	Perform SIM initialization with all the facilities supported		
	lacificies supported		
	1- Envelope menu selection with help		
	request is sent to the SIM	1- Applet1 is triggered	
	2- Applet1 gets the Proactive Handler		
		2- No exception is thrown	
		Applet1 finalizes	
4	Proactive Handler availability with EVENT_MENU_SELECTION		
	1- Envelope menu selection is sent to the	4. Appletd in triangular	
	SIM	1- Applet1 is triggered	
	2- Applet1 gets the Proactive Handler	2- No exception is thrown.	
		2- NO exception is tillown.	
		Applet1 finalizes	
		4 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	
5	Proactive Handler availability with		
	EVENT_FORMATTED_SMS_PP_ENV	1- Applet1 is triggered	
	1- Envelope dataDownLoad formatted is sent to the SIM		
	2- Applet1 gets the Proactive Handler	2- No exception is thrown.	
		Applet1 finalizes	

ld	Description	API /Framework Expectation	APDU Expectation
6	Proactive Handler availability with EVENT_UNFORMATTED_SMS_PP_ENV		
	1- Envelope dataDownLoad unformatted is sent to the SIM	1- Applet1 is triggered	
	2- Applet1 gets the Proactive Handler		
		2- No exception is thrown.	
	3- Applet2 gets the Proactive Handler	Applet1 finalizes Applet2 is triggered	
		3 No exception is thrown.	
7	Proactive Handler availability with EVENT_FORMATTED_CELL_BROADCAST		
	1- Envelope cell broadcast formatted is sent to the SIM	1- Applet1 is triggered	
	2- Applet1 gets the Proactive Handler		
		2-No exception is thrown	
		Applet1 finalizes	
8	Proactive Handler availability with EVENT_UNFORMATTED_CELL_BROADCAST		
	1- Envelope cell broadcast unformatted is sent to the SIM	1- Applet1 is triggered	
	2- Applet1 gets the Proactive Handler	2- No exception is thrown	
	3- Applet2 gets the Proactive Handler	Applet1 finalizes Applet2 is triggered	
		3 No exception is thrown	
9	Proactive Handler availability with EVENT_TIMER_EXPIRATION		
	1- Timer Id =1 Envelope Timer Expiration is sent to the SIM	1- Applet1 is triggered	
	2- Applet1 gets the Proactive Handler		
		2- No exception is thrown.	
		Applet1 finalizes	
10	Proactive Handler availability with EVENT_CALL_CONTROL_BY_SIM		
	1- Envelope call control by SIM is sent to the SIM	1- Applet1 is triggered	
	2- Applet1 gets the Proactive Handler	2- No exception is thrown.	

ld	Description	API /Framework Expectation	APDU Expectation
11	Proactive Handler availability with	·	-
	EVENT_MO_SHORT_MESSAGE_CONTROL		
	1- Envelope mo short message control by SIM is sent to the SIM	1- Applet1 is triggered	
	2- Applet1 gets the Proactive Handler	2- No exception is thrown	
12	Proactive Handler availability with EVENT_EVENT_DOWNLOAD_MT_CALL		
	1- Envelope event download mt call is sent to the SIM	1- Applet1 is triggered	
	2- Applet1 gets the Proactive Handler		
		2- No exception is thrown.	
	3- Applet2 gets the Proactive Handler	Applet1 finalizes	
		Applet2 is triggered	
		3-No exception is thrown	
13	Proactive Handler availability with EVENT_EVENT_DOWNLOAD_CALL_CONNECT ED		
	1- Envelope event download call connected is sent to the SIM	1- Applet1 is triggered	
	2- Applet1 gets the Proactive Handler	2- No exception is thrown.	
	3- Applet2 gets the Proactive Handler	Applet1 finalizes Applet2 is triggered	
		3- No exception is thrown	
14	Proactive Handler availability with EVENT_EVENT_DOWNLOAD_CALL_DISCONN ECTED		
	1- Envelope event download call disconnected is sent to the SIM	1- Applet1 is triggered	
	2- Applet1 gets the Proactive Handler		
	3- Applet2 gets the Proactive Handler	2- No exception is thrown.	
		Applet1 finalizes Applet2 is triggered	
		3- No exception is thrown.	

ld	Description	API /Framework Expectation	APDU Expectation
15	Applets triggering with	·	•
	EVENT_EVENT_LOCATION_STATUS		
	1- Envelope event download location status		
	is sent to the SIM	1- Applet1 is triggered	
	2- Applet1 gets the Proactive Handler		
		2- No exception is thrown.	
	3- Applet2 gets the Proactive Handler	Applet1 finalizes Applet2 is triggered	
16	Proactive Handler availability with	3- No exception is thrown	
10	EVENT_EVENT_DOWNLOAD_USER_ACTIVITY		
	1- Envelope event download user activity		
	is sent to the SIM	1- Applet1 is triggered	
	O Auglot 1 webs the Possetion World		
	2- Applet1 gets the Proactive Handler	2- No exception is thrown	
		Applet1 finalizes	
		Applet2 is triggered	
	3- Applet2 gets the Proactive Handler		
		3- No exception is thrown	
17	Proactive Handler availability with	•	
	EVENT_EVENT_DOWNLOAD_IDLE_SCREEN_ AVAILABLE		
	AVAILABLE		
	1- Envelope event download idle screen available is sent to the SIM	1- Applet1 is triggered	
		T- Applett is triggered	
	2- Applet1 gets the Proactive Handler		
	11 11 31 11 11 11 11 11 11 11		
		2- No exception is thrown.	
	3- Applet2 gets the Proactive Handler	Applet1 finalizes	
		Applet2 is triggered	
		3- No exception is thrown	
18	Proactive Handler availability with	o 140 exception is thrown	
	EVENT_EVENT_DOWNLOAD_CARD_READER _STATUS		
	1- Envelope event download card reader status is sent to the SIM	1- Applet1 is triggered	
	2- Applet1 gets the Proactive Handler		
	12 inspecting the invactive natural		
		2- No exception is thrown.	
	2 Applot 2 got g the Properties World and	Applet1 finalizes	
	3- Applet2 gets the Proactive Handler	Applet1 finalizes Applet2 is triggered	
		3- No exception is thrown	

ld	Description	API /Framework Expectation	APDU Expectation
19	Proactive Handler availability with EVENT_EVENT_DOWNLOAD_LANGUAGE_SE LECTION		
	1- Envelope event download language selection is sent to the SIM	1- Applet1 is triggered	
	2- Applet1 gets the Proactive Handler	2-No exception is thrown.	
	3- Applet2 gets the Proactive Handler	Applet1 finalizes Applet2 is triggered	
		3-No exception is thrown	
20	Proactive Handler availability with EVENT_EVENT_DOWNLOAD_BROWSER_TER MINATION	o tto oxoophorite tillowi	
	1- Envelope event download browser termination is sent to the SIM	1- Applet1 is triggered	
	2- Applet1 gets the Proactive Handler	2-No exception is thrown.	
		Applet1 finalizes	
	3- Applet2 gets the Proactive Handler	Applet2 is triggered	
		3-No exception is thrown	
21	Proactive Handler availability with EVENT_STATUS_COMMAND		
	1- Status command is sent to the SIM	1- Applet1 is triggered	
	2- Applet1 gets the Proactive Handler	2- No exception is thrown.	
		Applet1 finalizes	
	3- Applet2 gets the Proactive Handler	Applet2 is triggered	
		3- No exception is thrown.	
22	Proactive Handler availability with EVENT_EVENT_DOWNLOAD_DATA_AVAILAB LE		1- OPEN CHANNEL proactive Command is fetched
	<pre>1- Applet1 builds a proactive command OPEN CHANNEL proactiveHandler.send() method is called.</pre>	3-No exception is thrown.	TERMINAL RESPONSE is issued with Channel Id = 01
		Applet1 finalizes	
	2- An Envelope Event Download Data Available is sent to the SIM, with channelId=01.		
	3- Applet1 gets the Proactive Handler		
23	Proactive Handler availability with EVENT_EVENT_DOWNLOAD_CHANNEL_STAT US	1- Applet1 is triggered	
	1- An Envelope Event Download Channel Status is sent to the SIM, with ChannelId=01	2- No exception is thrown.	
	2- Applet1 gets the Proactive Handler	Applet1 finalizes	
		I	

ld	Description	API /Framework Expectation	APDU Expectation
24	Proactive Handler availability with UNRECOGNIZED_ENVELOPE 1- An unrecognized Envelope (BER TLV Tag unrecognized) is sent to the SIM	1- Applet1 is triggered	
	2- Applet1 gets the Proactive Handler	2- No exception is thrown.	
	3-Applet2 gets the Proactive Handler	Applet1 finalizes Applet2 is triggered 3- No exception is thrown	
25	Proactive Handler availability with EVENT_FORMATTED_SMS_PP_UPD		
	1- Update Record EFsms instruction formatted is sent to the SIM	1- Applet1 is triggered	
	2- Applet1 gets the Proactive Handler	2- No exception is thrown. Applet1 finalizes	
26	Proactive Handler availability with EVENT_UNFORMATTED_SMS_PP_UPD		
	1- Update Record EFsms instruction unformatted is sent to the SIM	1- Applet1 is triggered	
	1- Applet1 gets the Proactive Handler	2- No exception is thrown.	
	2- Applet2 gets the Proactive Handler	Applet1 finalizes 3- Applet2 is triggered	
		4- No exception is thrown.	

6.3.1.1.4 Test Coverage

CRR Number	Test Case Number		
CRRN1	2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21,		
	22, 23, 24, 25, 26		
CRRC1	1		

6.3.1.2 ProactiveResponseHandler

Test Area Reference: FWK_MHA_PRHD

6.3.1.2.1 Conformance Requirement

6.3.1.2.1.1 Normal Execution

• CRRN1: The ProactiveResponseHandler is available after the first call to the ProactiveHandler.send() method to the termination of the processToolkit method for the following events:

EVENT_FORMATTED_SMS_PP_ENV

EVENT_UNFORMATTED_SMS_PP_ENV

EVENT_FORMATTED_SMS_PP_UPD

EVENT_UNFORMATTED_SMS_PP_UPD

EVENT FORMATTED SMS CB

EVENT_UNFORMATTED_SMS_CB

EVENT_MENU_SELECTION

EVENT_MENU_SELECTION_HELP_REQUEST

EVENT_TIMER_EXPIRATION

EVENT_EVENT_DOWNLOAD_MT_CALL

EVENT_EVENT_DOWNLOAD_CALL_CONNECTED

EVENT_EVENT_DOWNLOAD_CALL_DISCONNECTED

EVENT_EVENT_DOWNLOAD_LOCATION_STATUS

EVENT_EVENT_DOWNLOAD_USER_ACTIVITY

EVENT_EVENT_DOWNLOAD_IDLE_SCREEN_AVAILABLE

EVENT_EVENT_DOWNLOAD_CARD_READER_STATUS

EVENT_EVENT_DOWNLOAD_LANGUAGE_SELECTION

EVENT_EVENT_DOWNLOAD_BROWSER_TERMINATION

EVENT_UNRECOGNIZED_ENVELOPE

EVENT_STATUS_COMMAND

EVENT_CALL_CONTROL

EVENT_SMS_MO_CONTROL

EVENT_PROFILE_DOWNLOAD

EVENT_EVENT_DOWNLOAD_DATA_AVAILABLE

EVENT_EVENT_DOWNLOAD_CHANNEL_STATUS

6.3.1.2.1.2 Parameters error

No requirements.

6.3.1.2.1.3 Context errors

• CRRC1: The ProactiveResponseHandler and its content are not available for any toolkit applet triggered from the invocation to the termination of their processToolkit method for the following events:

EVENT_FIRST_COMMAND_AFTER_SELECT

6.3.1.2.2 Test Suite Files

Test Script: FWK_MHA_PRHD_1.scr

Test Applet: FWK_MHA_PRHD_1.java

FWK_MHA_PRHD_2.java

Load Script: FWK_MHA_PRHD_1.ldr

Cleanup Script: FWK_MHA_PRHD_1.clr
Parameter File: FWK_MHA_PRHD_1.par

6.3.1.2.3 Test Procedure

	Description	API/Framework Expectation	APDU Expectation
1	Applets registration to all events and Proactive Response Handler availability with EVENT_PROFILE_DOWNLOAD		·
d E E E E E	applet1 is registered to all events defined in TS 43.019 [7] except WENT_FIRST_COMMAND_AFTER_SELECT, applet2 is registered to all events defined in TS 43.109[7] except WENT_FIRST_COMMAND_AFTER_SELECT, WENT_CALL_CONTROL_BY_SIM and WENT_MO_SMS_CONTROL_BY_SIM. desing the methods initMenuEntry() for WENT_MENU_SELECTION, dequestPollInterval() for WENT_STATUS_COMMAND, allocateTimer() for WENT_TIMER_EXPIRATION and setEventList() dor the rest of the events.		
S M	-Terminal Profile command is sent to the IM without the facility of ET_EVENT_LIST, POLL_INTERVAL, SET UP IDLE GODE TEXT and SET UP MENU.	1-Applet1 is triggered by EVENT_PROFILE_DOWNLOAD No exception is thrown	
3	PISPLAY TEXT ProactiveHandler.send() method is alled		3- The proactive comman DISPLAY TEXT is fetched
Ρ	roactiveResponseHandler.getTheHandler() ethod is called	4- No exception is thrown	TERMINAL RESPONSE
	pplet1 is deregistered to VENT_PROFILE_DOWNLOAD		
	- Applet2 builds a proactive command ISPLAY TEXT.	Applet1 finalizes	
	- ProactiveHandler.send() method is alled	Applet2 is triggered by EVENT_PROFILE_DOWNLOAD	
Ρ	roactiveResponseHandler.getTheHandler() method is called	7- No exception is thrown	6- The proactive comman DISPLAY TEXT is fetched TERMINAL RESPONSE

ld	Description	API/Framework Expectation	APDU Expectation
2	Proactive Response Handler availability with		•
	EVENT_MENU_SELECTION_HELP_REQUEST		
	Perform SIM initialization with all the		
	facilities supported		
	1-Envelope menu selection with help		
	request is sent to the SIM	1- Applet1 is triggered	
	Applet1 builds a proactive command DISPLAY		
	TEXT		
	2- ProactiveHandler.send() method is		2- A proactive command
	called		DISPLAY TEXT is fetched
			TERMINAL RESPONSE
	3- ProactiveResponseHandler.getTheHandler()		
	method is called	3- No exception is thrown	
3	Proactive Response Handler availability with EVENT_MENU_SELECTION		
	EVENT_MENO_OLLEGIION		
	1-Envelope menu selection is sent to the SIM	1- Applet1 is triggered	
	Applet1 builds a proactive command DISPLAY TEXT		
	ILAI		
	2- ProactiveHandler.send() method is called		2- A proactive command DISPLAY TEXT is fetched
	Called		DIOI LAT TEXT IS lettered
			TERMINAL RESPONSE
	3-ProactiveResponseHandler.getTheHandler()	3- No exception is thrown	
	method is called	10-140 exception is thrown	
4	Proactive Response Handler availability with		
	EVENT_FORMATTED_SMS_PP_ENV		
	1-Envelope dataDownLoad formatted is sent	1- Applet1 is triggered	
	to the SIM		
	Applet builds a proactive command DISPLAY		
	TEXT		2- A proactive command
	2-ProactiveHandler.send() method is called		DISPLAY TEXT is fetched
			TERMINAL RESPONSE
	2 Post of the Post of the Mark	3- No exception is thrown	
	3-ProactiveResponseHandler.getTheHandler() method is called		
<u></u>	Propositive Popularies House House 19 1 199		
5	Proactive Response Handler availability with EVENT_UNFORMATTED_SMS_PP_ENV		
	1-Envelope dataDownLoad unformatted is sent to the SIM	1- Applet1 is triggered	
	Applet1 builds a proactive command DISPLAY TEXT		
			2- A proactive command DISPLAY TEXT is fetched
	<pre>2- ProactiveHandler.send() method is called</pre>		DISPLAT TEAT IS TERCHED
			TERMINAL RESPONSE
	3-	3- No exception is thrown	
	ProactiveResponseHandler.getTheHandler()	140 evcebuon is unomi	
	method is called		
		Applet1 finalizes Applet2 is triggered	
		Lyphiers is midderen	
	Applet 2 builds a properties command DIGDLAY		
	Applet2 builds a proactive command DISPLAY		

ld	Description	API/Framework Expectation	APDU Expectation
	TEXT	P	
	4- ProactiveHandler.send() method is called		4- A proactive command DISPLAY TEXT is fetched
	5- ProactiveResponseHandler.getTheHandler() method is called	5- No exception is thrown	TERMINAL RESPONSE
6	Proactive Response Handler availability with EVENT_FORMATTED_SMS _CB		
	1-Envelope cell broadcast formatted is sent to the SIM	1- Applet1 is triggered	
	Applet1 builds a proactive command DISPLAY TEXT		
	2-ProactiveHandler.send() method is called		2- A proactive command DISPLAY TEXT is fetched
	3-ProactiveResponseHandler.getTheHandler() method is called.	3- No exception is thrown	TERMINAL RESPONSE
7	Proactive Response Handler availability with EVENT_UNFORMATTED_SMS _CB		
	1-Envelope call broadcast unformatted is sent to the SIM	1- Applet1 is triggered	
	Applet1 builds a proactive command DISPLAY TEXT		2- A proactive command
	2-ProactiveHandler.send() method is called		DISPLAY TEXT is fetched
	3-ProactiveResponseHandler.getTheHandler() method is called.	3- No exception is thrown	TERMINAL RESPONSE
		Applet1 finalizes Applet2 is triggered	
	Applet2 builds a proactive command DISPLAY TEXT	- · · · · · · · · · · · · · · · · · · ·	4. A propostive command
	4- ProactiveHandler.send() method is called		4- A proactive command DISPLAY TEXT is fetched
			TERMINAL RESPONSE
	5- ProactiveResponseHandler.getTheHandler() method is called	5- No exception is thrown	
8	Proactive Response Handler availability with EVENT_TIMER_EXPIRATION		
	Timer id=1 1-Envelope Timer Expiration is sent to the SIM	1- Applet1 is triggered	
	Applet builds a proactive command DISPLAY TEXT		
	2-ProactiveHandler.send() method is called		2- A proactive command DISPLAY TEXT is fetched
	3-ProactiveResponseHandler.getTheHandler() method is called	3- No exception is thrown	TERMINAL RESPONSE
9	Proactive Response Handler availability with EVENT_CALL_CONTROL_BY_SIM	1- Applet1 is triggered	
	1-Envelope call control by sim is sent to the SIM		
	Applet builds a proactive command DISPLAY TEXT		

ld	Description	API/Framework Expectation	APDU Expectation
	2-ProactiveHandler.send() method is called		2- A proactive command DISPLAY TEXT is fetched
	3-ProactiveResponseHandler.getTheHandler() method is called	3- No exception is thrown	TERMINAL RESPONSE
1	Proactive Response Handler availability with _ MO_SHORT_MESSAGE_CONTROL_BY_SIM		
	1-Envelope mo short message control by sim is sent to the SIM	1- Applet1 is triggered	
	Applet builds a proactive command DISPLAY TEXT		
	2-ProactiveHandler.send() method is called		2- A proactive command DISPLAY TEXT is fetched
	3-ProactiveResponseHandler.getTheHandler() method is called	3- No exception is thrown	TERMINAL RESPONSE
11	Proactive Response Handler availability with EVENT_EVENT_DOWNLOAD_MT_CALL		
	1-Envelope event download mt call is sent to the SIM	1- Applet1 is triggered	
	Applet1 builds a proactive command DISPLAY TEXT		
	2-ProactiveHandler.send() method is called		2- A proactive command DISPLAY TEXT is fetched
	3-ProactiveResponseHandler.getTheHandler() method is called.	3- No exception is thrown Applet1 finalizes	TERMINAL RESPONSE
	Applet2 builds a proactive command DISPLAY TEXT	Applet2 is triggered	
	4- ProactiveHandler.send() method is called		4- A proactive command DISPLAY TEXT is fetched
	5-		TERMINAL RESPONSE
	ProactiveResponseHandler.getTheHandler() method is called	5- No exception is thrown	

ld	Description	API/Framework Expectation	APDU Expectation
1	Proactive Response Handler availability with EVENT_EVENT_DOWNLOAD_CALL_CONNECT ED		
	1-Envelope event download call connected is sent to the SIM	1- Applet1 is triggered	
	Applet1 builds a proactive command DISPLAY TEXT		2- A proactive command DISPLAY TEXT is fetched
	2-ProactiveHandler.send() method is called		TERMINAL RESPONSE
	<pre>3-ProactiveResponseHandler.getTheHandler() method is called</pre>	3- No exception is thrown	
	Applet builds a proactive command DISPLAY	Applet1 finalizes Applet2 is triggered	
	4- ProactiveHandler.send() method is called		4- A proactive command DISPLAY TEXT is fetched
	5- ProactiveResponseHandler.getTheHandler() method is called	5- No exception is thrown	TERMINAL RESPONSE
13	Proactive Response Handler availability with EVENT_EVENT_DOWNLOAD_CALL_DISCONN ECTED		
	1-Envelope event download call disconnected is sent to the SIM	1- Applet1 is triggered	
	Applet1 builds a proactive command DISPLAY TEXT		
	2-ProactiveHandler.send() method is called		2- A proactive command DISPLAY TEXT is fetched
	3-ProactiveResponseHandler.getTheHandler() method is called	3- No exception is thrown	TERMINAL RESPONSE
		Applet1 finalizes Applet2 is triggered	
	Applet2 builds a proactive command DISPLAY TEXT		
	4- ProactiveHandler.send() method is called		4- A proactive command DISPLAY TEXT is fetched
	5- ProactiveResponseHandler.getTheHandler() method is called	5- No exception is thrown	TERMINAL RESPONSE
1	Proactive Response Handler availability with EVENT_EVENT_DOWNLOAD_LOCATION_STA TUS		
	1-Envelope event download location status is sent to the SIM	1- Applet1 is triggered	
	Applet1 builds a proactive command DISPLAY TEXT		2-A proactive command
	2-ProactiveHandler.send() method is called		DISPLAY TEXT is fetched TERMINAL RESPONSE
		<u>l</u>	I LIKIMINAL KLOI ONGE

ld	Description	API/Framework Expectation	APDU Expectation
	3-ProactiveResponseHandler.getTheHandler() method is called	3- No exception is thrown	
	method is carred	Applet1 finalizes Applet2 is triggered	
	Applet2 builds a proactive command DISPLAY TEXT		
	4- ProactiveHandler.send() method is called		4- A proactive command DISPLAY TEXT is fetched
	5- ProactiveResponseHandler.getTheHandler() method is called	5- No exception is thrown	TERMINAL RESPONSE
15	Proactive Response Handler availability with EVENT_EVENT_DOWNLOAD_USER_ACTIVITY		
	1-Envelope event download user activity is sent to the SIM	1- Applet1 is triggered	
	Applet1 builds a proactive command DISPLAY TEXT		
	2-ProactiveHandler.send() method is called		2-A proactive command DISPLAY TEXT is fetched
	3-ProactiveResponseHandler.getTheHandler() method is called	3- No exception is thrown	TERMINAL RESPONSE
	Applet2 builds a proactive command DISPLAY	Applet1 finalizes Applet2 is triggered	
	TEXT 4- ProactiveHandler.send() method is called		4- A proactive command DISPLAY TEXT is fetched
	called		TERMINAL RESPONSE
	5- ProactiveResponseHandler.getTheHandler() method is called	5- No exception is thrown	
16	Proactive Response Handler availability with EVENT_EVENT_DOWNLOAD_IDLE_SCREEN_ AVAILABLE		
	1-Envelope event download idle screen available is sent to the SIM	1- Applet1 is triggered	
	Applet1 builds a proactive command DISPLAY TEXT		
	2-ProactiveHandler.send() method is called		2- A proactive command DISPLAY TEXT is fetched
	3-ProactiveResponseHandler.getTheHandler() method is called	3- No exception is thrown	TERMINAL RESPONSE
		Applet1 finalizes Applet2 is triggered	
	Applet2 builds a proactive command DISPLAY TEXT		
	4- ProactiveHandler.send() method is called		4- A proactive command DISPLAY TEXT is fetched
	5-		TERMINAL RESPONSE
17	ProactiveResponseHandler.getTheHandler() method is called	5- No exception is thrown	
17	Proactive Response Handler availability with		

ld	Description	API/Framework Expectation	APDU Expectation
	EVENT_EVENT_DOWNLOAD_CARD_READER		'
	_STATUS		
	1-Envelope event download card reader status is sent to the SIM	1- Applet1 is triggered	
	second 15 Sent to the Sin		
	Applet1 builds a proactive command DISPLAY		
	TEXT		
	2-ProactiveHandler.send() method is called		2-A proactive command
			DISPLAY TEXT is fetched
			TERMINAL RESPONSE
	3-ProactiveResponseHandler.getTheHandler()	3- No exception is thrown	TERMINAL RESPONSE
	method is called	The exception is the time.	
		Applet1 finalizes	
		Applet2 is triggered	
	Applet2 builds a proactive command DISPLAY		
	TEXT		
	4- ProactiveHandler.send() method is		
	called		4- A proactive command
			DISPLAY TEXT is fetched
	5-		TERMINAL RESPONSE
	ProactiveResponseHandler.getTheHandler()		TERMINAL RESPONSE
	method is called	5- No exception is thrown	
18	Proactive Response Handler availability with		
	EVENT_EVENT_DOWNLOAD_LANGUAGE_		
	SELECTION		
	1-Envelope event download language	1- Applet1 is triggered	
	selection is sent to the SIM	, , , pp.ot. io inggores	
	Applet1 builds a proactive command DISPLAY		
	TEXT		
			2-A proactive command
	2-ProactiveHandler.send() method is called		DISPLAY TEXT is fetched
			TERMINAL RESPONSE
	3-ProactiveResponseHandler.getTheHandler() method is called	O No constitution is the	
	method is called	3-No exception is thrown	
		Applet1 finalizes	
		Applet2 is triggered	
	Applet2 builds a proactive command DISPLAY		
1	TEXT		
	4- ProactiveHandler.send() method is		
	called		4-A proactive command
			DISPLAY TEXT is fetched
	5-		
	ProactiveResponseHandler.getTheHandler()	5-No exception is thrown	TERMINAL RESPONSE
	method is called		

ld	Description	API/Framework Expectation	APDU Expectation
19	Proactive Response Handler availability with EVENT_EVENT_DOWNLOAD_BROWSER_ TERMINATION		
	1-Envelope event download Browser termination is sent to the SIM	1- Applet1 is triggered	
	Applet1 builds a proactive command DISPLAY TEXT		
	2-ProactiveHandler.send() method is called		2-A proactive command DISPLAY TEXT is fetched
	3-ProactiveResponseHandler.getTheHandler() method is called	3-No exception is thrown	TERMINAL RESPONSE
	Applet2 builds a proactive command DISPLAY	Applet1 finalizes Applet2 is triggered	
	TEXT		
	<pre>4- ProactiveHandler.send() method is called 5-</pre>		4-A proactive command DISPLAY TEXT is fetched
	ProactiveResponseHandler.getTheHandler() method is called	5-No exception is thrown	TERMINAL RESPONSE
20	Proactive Response Handler availability with EVENT_STATUS_COMMAND		
	1-Status command is sent to the SIM	1- Applet1 is triggered	
	Applet1 builds a proactive command DISPLAY TEXT		
	2-ProactiveHandler.send() method is called		2- A proactive command DISPLAY TEXT is fetched
	3-ProactiveResponseHandler.getTheHandler() method is called	3- No exception is thrown	TERMINAL RESPONSE
		Applet1 finalizes Applet2 is triggered	
	Applet2 builds a proactive command DISPLAY TEXT		
	4- ProactiveHandler.send() method is called		4- A proactive command DISPLAY TEXT is fetched
	5-		TERMINAL RESPONSE
21	Proactive Handler .getTheHandler() method is called	5- No exception is thrown	
21	Proactive Handler availability with EVENT_EVENT_DOWNLOAD_DATA_AVAILAB LE		
	1- Applet1 builds a proactive command OPEN CHANNEL. proactiveHandler.send() method is called	1- Applet1 is registered to EVENT_EVENT_DOWNLOAD_DA TA_AVAILABLE and EVENT_EVENT_DOWNLOAD_CH ANNEL_STATUS	1- OPEN CHANNEL proactive command is fetched TERMINAL RESPONSE is
	2- An Envelope Event Download Data Available is sent to the SIM, with ChannelId=01.	2- Applet1 is triggered	issued with Channel Id = 01
	3-Applet1 builds a proactive command DISPLAY TEXT		

ld	Description	API/Framework Expectation	APDU Expectation
	4- ProactiveHandler.send() method is called		4- A proactive command DISPLAY TEXT is fetched
			TERMINAL RESPONSE
	5- ProactiveResponseHandler.getTheHandler() method is called	5- No exception is thrown	

ld	Description	API/Framework Expectation	APDU Expectation
22	Proactive Handler availability with	Al III Tamework Expectation	Al Do Expectation
	EVENT_EVENT_DOWNLOAD_CHANNEL_STAT US	1- Applet1 is triggered	
	1-An Envelope Event Download Channel Status is sent to the SIM with ChannelId=01.		
	Applet1 builds a proactive command DISPLAY TEXT		
	2- ProactiveHandler.send() method is called		2- A proactive command DISPLAY TEXT is fetched
	3- ProactiveResponseHandler.getTheHandler() method is called	3- No exception is thrown	TERMINAL RESPONSE
23	Proactive Response Handler availability with UNRECOGNIZED_ENVELOPE		
	1-An unrecognized Envelope is sent to the SIM	1- Applet1 is triggered	
	Applet1 builds a proactive command DISPLAY TEXT		
	2-ProactiveHandler.send() method is called		2- A proactive command DISPLAY TEXT is fetched
	3-ProactiveResponseHandler.getTheHandler() method is called	3- No exception is thrown Applet1 finalizes	TERMINAL RESPONSE
	Applet2 builds a proactive command DISPLAY TEXT	Applet2 is triggered	
	4- ProactiveHandler.send() method is called		4- A proactive command DISPLAY TEXT is fetched
	5- ProactiveResponseHandler.getTheHandler() method is called	5- No exception is thrown	TERMINAL RESPONSE
24	Proactive Response Handler availability with EVENT_FORMATTED_SMS_PP_UPD		
	1- Update Record EFsms instruction formatted is sent to the SIM	1- Applet1 is triggered	
	Applet builds a proactive command DISPLAY TEXT		2- A proactive command
	2-ProactiveHandler.send() method is called		DISPLAY TEXT is fetched
	method is called	3- No exception is thrown	TERMINAL RESPONSE
25	Proactive Response Handler availability with EVENT_UNFORMATTED_SMS_PP_UPD		
	1- Update Record EFsms instruction unformatted is sent to the SIM	1- Applet1 is triggered	
	Applet1 builds a proactive command DISPLAY TEXT		

Description	API/Framework Expectation	APDU Expectation
2- ProactiveHandler.send() method is called		2- A proactive command DISPLAY TEXT is fetched
3- ProactiveResponseHandler.getTheHandler() method is called	3- No exception is thrown Applet1 finalizes Applet2 is triggered	TERMINAL RESPONSE
TEXT 4- ProactiveHandler.send() method is		4- A proactive command
5- ProactiveResponseHandler.getTheHandler()	5- No exception is thrown	DISPLAY TEXT is fetched TERMINAL RESPONSE
	2- ProactiveHandler.send() method is called 3- ProactiveResponseHandler.getTheHandler() method is called Applet2 builds a proactive command DISPLAY TEXT 4- ProactiveHandler.send() method is called 5-	2- ProactiveHandler.send() method is called 3- ProactiveResponseHandler.getTheHandler() method is called Applet1 finalizes Applet2 is triggered Applet2 builds a proactive command DISPLAY TEXT 4- ProactiveHandler.send() method is called 5- ProactiveResponseHandler.getTheHandler() 5- No exception is thrown

6.3.1.2.4 Test Coverage

CRR Number	Test Case Number	
CRRN1	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20,	
	21, 22, 23, 24, 25	
CRRC1	Not testable	

6.3.1.3 EnvelopeHandler

Test Area Reference: FWK_MHA_ENHD

6.3.1.3.1 Conformance Requirement

6.3.1.3.1.1 Normal Execution

• CRRN1: The EnvelopeHandler and its content are available for all toolkit applets triggered from the invocation to the termination of their processToolkit method for the following events:

EVENT_FORMATTED_SMS_PP_ENV

EVENT_UNFORMATTED_SMS_PP_ENV

EVENT_FORMATTED_SMS_PP_UPD

EVENT_UNFORMATTED_SMS_PP_UPD

EVENT_FORMATTED_SMS_CB

EVENT_UNFORMATTED_SMS_CB

EVENT_MENU_SELECTION

EVENT_MENU_SELECTION_HELP_REQUEST

EVENT_TIMER_EXPIRATION

 ${\tt EVENT_EVENT_DOWNLOAD_MT_CALL}$

EVENT_EVENT_DOWNLOAD_CALL_CONNECTED

EVENT_EVENT_DOWNLOAD_CALL_DISCONNECTED

EVENT_EVENT_DOWNLOAD_LOCATION_STATUS

EVENT_EVENT_DOWNLOAD_USER_ACTIVITY

EVENT_EVENT_DOWNLOAD_IDLE_SCREEN_AVAILABLE

EVENT_EVENT_DOWNLOAD_CARD_READER_STATUS

EVENT_EVENT_DOWNLOAD_LANGUAGE_SELECTION

EVENT_EVENT_DOWNLOAD_BROWSER_TERMINATION

EVENT_UNRECOGNIZED_ENVELOPE

EVENT_CALL_CONTROL

EVENT_SMS_MO_CONTROL

EVENT_EVENT_DOWNLOAD_DATA_AVAILABLE

EVENT_EVENT_DOWNLOAD_CHANNEL_STATUS

6.3.1.3.1.2 Parameters error

No requirements.

6.3.1.3.1.3 Context Errors

• CRRC1: The EnvelopeHandler and its content are not available for any toolkit applet triggered from the invocation to the termination of their processToolkit method for the following events:

EVENT_STATUS_COMMAND

EVENT_PROFILE_DOWNLOAD

EVENT_FIRST_COMMAND_AFTER_SELECT

6.3.1.3.2 Test Suite Files

Test Script: FWK_MHA_ENHD_1.scr

Test Applet: FWK_MHA_ENHD_1.java

FWK_MHA_ENHD_2.java

Load Script: FWK_MHA_ENHD_1.ldr

Cleanup Script: FWK_MHA_ENHD_1.clr

Parameter File: FWK_MHA_ENHD_1.par

6.3.1.3.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1	Applet1 and Applet2 registration and Envelope Handler availability with EVENT_FIRST_COMMAND_AFTER_SELECT		·
	1.Applet1 is registered to all events defined TS 43.019 [7]. The registration is done using the methods initMenuEntry() for EVENT_MENU_SELECTION, requestPollInterval() for EVENT_STATUS_COMMAND, allocateTimer() for EVENT_TIMER_EXPIRATION and setEventList() for the rest of the events.	1- No exception is thrown	
	Applet2 is registered to all events defined TS 43.019 [7] except EVENT_PROFILE_DOWNLOAD, EVENT_CALL_CONTROL_BY_SIM and EVENT_MO_SHORT_MESSAGE_CONTROL_BY_SIM. The registration is done using the methods initMenuEntry() for EVENT_MENU_SELECTION, requestPollInterval() for EVENT_STATUS_COMMAND, allocateTimer for EVENT_TIMER_EXPIRATION and setEventList for the rest of the events.	2- Applet1 is triggered by EVENT_FIRST_COMMAND_AFTE R_SELECT	
	2- Select MF.		
	3-EnvelopeHandler.getTheHandler() method is called by Applet1 Applet1 is deregistered from EVENT_FIRST_COMMAND_AFTER_SELECT. 4-EnvelopeHandler.getTheHandler() method is called by Applet2	3- A Toolkit exception HANDLER_NOT_AVAILABLE is thrown Applet1 finalizes Applet2 is triggered	
	Applet2 is deregistered to EVENT_FIRST_COMMAND_AFTER_SELECT.	4- A Toolkit exception HANDLER_NOT_AVAILABLE is thrown	
2	Handler availability with EVENT_PROFILE_DOWNLOAD		
	1- Terminal Profile command is sent to the SIM without the facility of SET_EVENT_LIST, SETUP_IDLE_MODE_TEXT, POLL_INTERVAL and SETUP MENU	1- Applet1 is triggered by EVENT_PROFILE_DOWNLOAD 2- A Toolkit exception	
	2- EnvelopeHandler.getTheHandler() method is called by Applet1	HANDLER_NOT_AVAILABLE is thrown	
	Applet1 is deregistered to EVENT_PROFILE_DOWNLOAD	Applet1 finalizes	
	3-EnvelopeHandler.getTheHandler() method is called by Applet2 Applet2 is deregistered to	Applet2 is triggered by EVENT_PROFILE_DOWNLOAD	
	EVENT_PROFILE_DOWNLOAD	3- A Toolkit exception HANDLER_NOT_AVAILABLE is thrown	

ld	Description	API/Framework Expectation	APDU Expectation
3	Envelope Handler availability with	p	
	EVENT_MENU_SELECTION_HELP_REQUEST		
	Perform SIM initialization with all the facilities supported		
	racificies supported		
	Envelope menu selection with help request	1- Applet1 is triggered	
	is sent to the SIM		
		2. No expension is thrown	
	1-EnvelopeHandler.getTheHandler() method is called by Applet1	2- No exception is thrown.	
	is carred by Appreci		
4	Envelope Handler availability with		
	EVENT_MENU_SELECTION		
	1-Envelope menu selection is sent to the	1 Applett is triggered	
	SIM	1- Applet1 is triggered	
	2-EnvelopeHandler.getTheHandler() method	2- No exception is thrown.	
	is called by Applet1		
5	Envelope Handler availability with		
	EVENT_FORMATTED_SMS_PP_ENV		
	1-A EVENT_FORMATTED_SMS_PP_ENV envelope is	1- Applet1 is triggered	
	sent to the SIM	7 Applet 13 triggered	
	2-EnvelopeHandler.getTheHandler() method	2- No exception is thrown.	
	is called by Applet1		
6	Envelope Handler availability with EVENT_UNFORMATTED_SMS_PP_ENV		
	EVENT_UNFORMATTED_SMS_PP_ENV		
	1-An unformatted sms pp envelope is sent	1- Applet1 is triggered	
	to the SIM		
	2-EnvelopeHandler.getTheHandler() method is called by Applet1	2- No exception is thrown.	
	* ** **		
		Applet1 finalizes	
		3- Applet2 is triggered	
	3-EnvelopeHandler.getTheHandler() method		
	is called by Applet2		
		4- No exception is thrown.	
7	Envelope Handler availability with		
	EVENT_FORMATTED_CB		
	1-Envelope cell broadcast formatted is		
	sent to the SIM	1- Applet1 is triggered	
	2-EnvelopeHandler.getTheHandler() method	2-No exception is thrown	
8	is called by Applet1 Envelope Handler availability with		
	EVENT_UNFORMATTED_CB		
	1-Envelope cell broadcast unformatted is		
1	1 DIVELOPE CELL DIDAGEASE UILLOIMALLEG IS	Î.	

ld	Description	API/Framework Expectation	APDU Expectation
	sent to the SIM	1- Applet1 is triggered	
	2-EnvelopeHandler.getTheHandler() method is called by Applet1	2- No exception is thrown	
	2. The second se	Applet1 finalizes	
	3-EnvelopeHandler.getTheHandler() method is called by Applet2	3- Applet2 is triggered	
		4- No exception is thrown	
9	Envelope Handler availability with EVENT_TIMER_EXPIRATION		
	Timer id=1 1-Envelope Timer Expiration is sent to the SIM	1- Applet1 is triggered	
	2-EnvelopeHandler.getTheHandler() method is called by Applet1	2- No exception is thrown.	
10	Envelope Handler availability with EVENT_CALL_CONTROL_BY_SIM		
	1-Envelope call control by sim is sent to the SIM	1- Applet1 is triggered	
	2-EnvelopeHandler.getTheHandler() method is called by Applet1	2- No exception is thrown.	
11	Envelope Handler availability with EVENT_MO_SHORT_MESSAGE_CONTROL_B Y_SIM		
	1-Envelope mo short message control by sim is sent to the SIM	1- Applet1 is triggered	
	2-EnvelopeHandler.getTheHandler() method is called by Applet1.	2- No exception is throw	
12	Envelope Handler availability with EVENT_EVENT_DOWNLOAD_MT_CALL		
	1-Envelope event download mt call is sent to the SIM	1- Applet1 is triggered	
	2-EnvelopeHandler.getTheHandler() method is called by Applet1	2- No exception is thrown.	
		Applet1 finalizes	
	3-EnvelopeHandler.getTheHandler() method is called by Applet2	3- Applet2 is triggered	
		4- No exception is thrown.	
13	Envelope Handler availability with EVENT_EVENT_DOWNLOAD_CALL_CONNECT ED		
	1-Envelope event download call connected is sent to the SIM	1- Applet1 is triggered	
	2-EnvelopeHandler.getTheHandler() method is called by Applet1	2- No exception is thrown.	
	3-EnvelopeHandler.getTheHandler() method is called by Applet2	Applet1 finalizes	
		3- Applet2 is triggered	

## A No exception is thrown. ## EVENT_EVENT_DOWNLOAD_CALL_DISCONE CTTED 1-Envelope event download call disconnected is sent to the SIN 2-EnvelopeRandler.getTheRandler() method is called by Applet1 15	ld	Description	API/Framework Expectation	APDU Expectation
Envelope Handler availability with EVENT_EVENT_DOWNLOAD_CALL_DISCONE			4- No exception is thrown.	
disconnected is sent to the SIM 2-EnvelopeBandler.getTheHandler() method is called by Applet1 3-EnvelopeBandler.getTheHandler() method is called by Applet2 5-Envelope Handlor availability with EVENT_EVENT_DOWNLOAD_LOCATION_STATUS 1-EnvelopeBandler.getTheHandler() method is called by Applet2 3-EnvelopeBandler.getTheHandler() method is called by Applet1 6-Envelope Handlor availability with EVENT_EVENT_DOWNLOAD_USER_ACTIVITY 1-Envelope Handlor availability with EVENT_EVENT_DOWNLOAD_USER_ACTIVITY 1-Envelope event download user activity is sent to the SIM 2-EnvelopeBandler.getTheHandler() method is called by Applet2 3-EnvelopeBandler.getTheHandler() method is called by Applet2 1-EnvelopeBandler.getTheHandler() method is called by Applet2 2-No exception is thrown. Applet1 finalizes 3-Applet2 is triggered 4-No exception is thrown. Applet1 finalizes 3-Applet2 is triggered 4-No exception is thrown.	14	EVENT_EVENT_DOWNLOAD_CALL_DISCONE	·	
is called by Applet1 3-EnvelopeHandler.getTheBandler() method is called by Applet2 4- No exception is thrown. Envelope Handler availability with EVENT_EVENT_DOWNLOAD_LOCATION_STA TUS 1-EnvelopeHandler.getTheBandler() method is called by Applet1 2-EnvelopeHandler.getTheBandler() method is called by Applet2 2-EnvelopeHandler.getTheBandler() method is called by Applet2 3-EnvelopeHandler availability with EVENT_EVENT_DOWNLOAD_USER_ACTIVITY 1-Envelope Handler availability with EVENT_EVENT_DOWNLOAD_USER_ACTIVITY 1-Envelope avent download user activity is sent to the SIM 2-EnvelopeHandler.getTheBandler() method is called by Applet1 2-EnvelopeHandler.getTheBandler() method is called by Applet2 4- No exception is thrown Applet1 finalizes 3-EnvelopeHandler.getTheBandler() method is called by Applet2 4- No exception is thrown Applet2 is triggered 4- No exception is thrown Applet3 is triggered 4- No exception is thrown Applet4 finalizes 3- Applet5 is triggered 4- No exception is thrown Applet6 is triggered 4- No exception is thrown Applet7 finalizes 3- Applet8 is triggered 4- No exception is thrown Applet1 is triggered 4- No exception is thrown. Applet1 is triggered 4- No exception is thrown. Applet1 finalizes 3- Applet2 is triggered 4- No exception is thrown.			1- Applet1 is triggered.	
3-EnvelopeHandler.getTheHandler() method is called by Applet1 15 Envelope Handler availability with EVENT_EVENT_DOWNLOAD_LOCATION_STA TUS 1-Envelope event download location status is sent to the SIM 2-EnvelopeHandler.getTheHandler() method is called by Applet1 3-RenvelopeHandler.getTheHandler() method is called by Applet2 4- No exception is thrown. Applet1 is triggered 4- No exception is thrown. Applet2 is triggered 4- No exception is thrown. Envelope Handler availability with EVENT_EVENT_DOWNLOAD_USER_ACTIVITY 1-Envelope event download user activity is sent to the SIM 2-EnvelopeHandler.getTheHandler() method is called by Applet2 3- Applet2 is triggered 4- No exception is thrown. 2- No exception is thrown Applet1 finalizes 3- Applet2 is triggered 4- No exception is thrown Applet1 finalizes 3- Applet2 is triggered 4- No exception is thrown Applet1 finalizes 3- Applet2 is triggered 4- No exception is thrown Applet1 finalizes 3- Applet2 is triggered 4- No exception is thrown Envelope Handler availability with EVENT_EVENT_DOWNLOAD_IDLE_SCREEN_AVAILABLE 1-Envelope event download idle screen available is sent to the SIM 1- Applet1 is triggered 4- No exception is thrown Applet1 is triggered 4- No exception is thrown Applet1 is triggered 4- No exception is thrown. Applet1 is triggered 4- No exception is thrown. Applet1 is triggered 4- No exception is thrown.				
Envelope Handler availability with EVENT_EVENT_DOWNLOAD_LOCATION_STA TUS 1- Applet1 is triggered 2- EnvelopeHandler.getTheHandler() method is called by Applet2 3- Applet2 is triggered 4- No exception is thrown. Applet1 finalizes 3- Applet2 is triggered 4- No exception is thrown. Applet1 finalizes 3- Applet2 is triggered 4- No exception is thrown. Applet1 finalizes 3- Applet2 is triggered 4- No exception is thrown. Applet1 finalizes 3- Applet2 is triggered 4- No exception is thrown. Applet1 finalizes 3- EnvelopeHandler.getTheHandler() method is called by Applet2 4- No exception is thrown Applet1 finalizes 3- EnvelopeHandler.getTheHandler() method is called by Applet2 4- No exception is thrown Applet1 finalizes 3- Applet2 is triggered 4- No exception is thrown Applet1 finalizes 3- Applet2 is triggered 4- No exception is thrown Applet1 finalizes 3- Applet1 is triggered 4- No exception is thrown Applet1 finalizes 3- Applet1 is triggered 4- No exception is thrown Applet1 finalizes 3- Applet1 is triggered 4- No exception is thrown Applet1 finalizes 3- Applet1 is triggered 4- No exception is thrown Applet1 finalizes 3- Applet2 is triggered 4- No exception is thrown Applet1 finalizes 3- Applet2 is triggered 4- No exception is thrown Applet1 finalizes 3- Applet2 is triggered 4- No exception is thrown Applet1 finalizes 3- Applet2 is triggered 4- No exception is thrown Applet1 finalizes 3- Applet2 is triggered 4- No exception is thrown Applet1 finalizes 3- Applet2 is triggered 4- No exception is thrown Applet1 finalizes 3- Applet2 is triggered 4- No exception is thrown Applet1 finalizes 3- Applet2 is triggered 4- No exception is thrown Applet1 finalizes 3- Applet2 is triggered 4- No exception is thrown Applet1 finalizes 3- Applet2 i				
Envelope Handler availability with EVENT_EVENT_DOWNLOAD_LOCATION_STA TUS 1- Applet1 is triggered 2- EnvelopeHandler.getTheHandler() method is called by Applet2 3- Applet2 is triggered 4- No exception is thrown. Applet1 finalizes 3- Applet2 is triggered 4- No exception is thrown. Applet1 finalizes 3- Applet2 is triggered 4- No exception is thrown. Applet1 finalizes 3- Applet2 is triggered 4- No exception is thrown. Applet1 finalizes 3- Applet2 is triggered 4- No exception is thrown. Applet1 finalizes 3- EnvelopeHandler.getTheHandler() method is called by Applet2 4- No exception is thrown Applet1 finalizes 3- EnvelopeHandler.getTheHandler() method is called by Applet2 4- No exception is thrown Applet1 finalizes 3- Applet2 is triggered 4- No exception is thrown Applet1 finalizes 3- Applet2 is triggered 4- No exception is thrown Applet1 finalizes 3- Applet1 is triggered 4- No exception is thrown Applet1 finalizes 3- Applet1 is triggered 4- No exception is thrown Applet1 finalizes 3- Applet1 is triggered 4- No exception is thrown Applet1 finalizes 3- Applet1 is triggered 4- No exception is thrown Applet1 finalizes 3- Applet2 is triggered 4- No exception is thrown Applet1 finalizes 3- Applet2 is triggered 4- No exception is thrown Applet1 finalizes 3- Applet2 is triggered 4- No exception is thrown Applet1 finalizes 3- Applet2 is triggered 4- No exception is thrown Applet1 finalizes 3- Applet2 is triggered 4- No exception is thrown Applet1 finalizes 3- Applet2 is triggered 4- No exception is thrown Applet1 finalizes 3- Applet2 is triggered 4- No exception is thrown Applet1 finalizes 3- Applet2 is triggered 4- No exception is thrown Applet1 finalizes 3- Applet2 is triggered 4- No exception is thrown Applet1 finalizes 3- Applet2 i			4- No exception is thrown.	
is sent to the SIM 2-EnvelopeHandler.getTheHandler() method is called by Applet1 3-EnvelopeHandler.getTheHandler() method is called by Applet2 4- No exception is thrown. Envelope Handler availability with EVENT_EVENT_DOWNLOAD_USER_ACTIVITY 1-Envelope event download user activity is sent to the SIM 2-EnvelopeHandler.getTheHandler() method is called by Applet1 3-EnvelopeHandler.getTheHandler() method is called by Applet2 4- No exception is thrown Applet1 is triggered 2- No exception is thrown Applet1 is triggered 2- No exception is thrown Applet1 is triggered 4- No exception is thrown Applet1 is triggered 2- No exception is thrown Applet1 is triggered 1- Applet1 is triggered 2- No exception is thrown 17 Envelope Handler availability with EVENT_EVENT_DOWNLOAD_IDLE_SCREEN_AVAILABLE 1-Envelope event download idle screen available is sent to the SIM 2-EnvelopeHandler.getTheHandler() method is called by Applet1 2- No exception is thrown. Applet1 finalizes 3- Applet2 is triggered 4- No exception is thrown. Applet1 finalizes 3- Applet2 is triggered 4- No exception is thrown. Applet1 finalizes 3- Applet2 is triggered	15	EVENT_EVENT_DOWNLOAD_LOCATION_STA		
is called by Applet1 3-EnvelopeHandler.getTheHandler() method is called by Applet2 4- No exception is thrown. Envelope Handler availability with EVENT_EVENT_DOWNLOAD_USER_ACTIVITY 1-Envelope event download user activity is sent to the SIM 2-EnvelopeHandler.getTheHandler() method is called by Applet1 2-EnvelopeHandler.getTheHandler() method is called by Applet2 3-EnvelopeHandler availability with EVENT_EVENT_DOWNLOAD_IDLE_SCREEN_AVAILABLE 1-Envelope event download idle screen available is sent to the SIM 2-EnvelopeHandler.getTheHandler() method is called by Applet1 2-EnvelopeHandler.getTheHandler() method is called by Applet2 4- No exception is thrown. Applet1 finalizes 3-Applet2 is triggered 4- No exception is thrown. Applet1 finalizes 3-Applet2 is triggered 4- No exception is thrown.			1- Applet1 is triggered	
3-EnvelopeHandler.getTheHandler() method is called by Applet2 4- No exception is thrown. 16 Envelope Handler availability with EVENT_EVENT_DOWNLOAD_USER_ACTIVITY 1-Envelope event download user activity is sent to the SIM 2-EnvelopeHandler.getTheHandler() method is called by Applet1 2- No exception is thrown Applet1 finalizes 3-Applet2 is triggered 2- No exception is thrown Applet1 finalizes 3-Applet2 is triggered 4- No exception is thrown 17 Envelope Handler availability with EVENT_EVENT_DOWNLOAD_IDLE_SCREEN_AVAILABLE 1-Envelope event download idle screen available is sent to the SIM 2-EnvelopeHandler.getTheHandler() method is called by Applet1 2- No exception is thrown. Applet1 is triggered 2- No exception is thrown. Applet1 finalizes 3- Applet2 is triggered 4- No exception is thrown. Applet1 finalizes 3- Applet2 is triggered 4- No exception is thrown.			2- No exception is thrown.	
### A No exception is thrown. ### Envelope Handler availability with EVENT_EVENT_DOWNLOAD_USER_ACTIVITY -Envelope event download user activity is sent to the SIM 1- Applet1 is triggered -EnvelopeHandler.getTheHandler() method is called by Applet1 2- No exception is thrown				
Envelope Handler availability with EVENT_EVENT_DOWNLOAD_USER_ACTIVITY 1-Envelope event download user activity is sent to the SIM 1- Applet1 is triggered 2-EnvelopeHandler.getTheHandler() method is called by Applet1 2- No exception is thrown Applet1 finalizes 3- EnvelopeHandler.getTheHandler() method is called by Applet2 4- No exception is thrown 17				
1-Envelope event download user activity is sent to the SIM 2-EnvelopeHandler.getTheHandler() method is called by Applet1 2- No exception is thrown Applet1 finalizes 3-EnvelopeHandler.getTheHandler() method is called by Applet2 4- No exception is thrown 17 Envelope Handler availability with EVENT_EVENT_DOWNLOAD_IDLE_SCREEN_AVAILABLE 1-Envelope event download idle screen available is sent to the SIM 2-EnvelopeHandler.getTheHandler() method is called by Applet1 2- No exception is thrown. Applet1 is triggered 2- No exception is thrown. Applet1 finalizes 3- Applet2 is triggered 4- No exception is thrown.	16		·	
is called by Applet1 2- No exception is thrown Applet1 finalizes 3-EnvelopeHandler.getTheHandler() method is called by Applet2 4- No exception is thrown 17 Envelope Handler availability with EVENT_EVENT_DOWNLOAD_IDLE_SCREEN_ AVAILABLE 1-Envelope event download idle screen available is sent to the SIM 2-EnvelopeHandler.getTheHandler() method is called by Applet1 2- No exception is thrown. Applet1 is triggered 2- No exception is thrown. Applet1 finalizes 3- Applet2 is triggered 4- No exception is thrown.		1-Envelope event download user activity is		
3-EnvelopeHandler.getTheHandler() method is called by Applet2 4- No exception is thrown 17 Envelope Handler availability with EVENT_EVENT_DOWNLOAD_IDLE_SCREEN_AVAILABLE 1-Envelope event download idle screen available is sent to the SIM 2-EnvelopeHandler.getTheHandler() method is called by Applet1 2- No exception is thrown. Applet1 finalizes 3- Applet2 is triggered 4- No exception is thrown.				
is called by Applet2 4- No exception is thrown 17 Envelope Handler availability with EVENT_EVENT_DOWNLOAD_IDLE_SCREEN_AVAILABLE 1-Envelope event download idle screen available is sent to the SIM 2-EnvelopeHandler.getTheHandler() method is called by Applet1 2- No exception is thrown. Applet1 finalizes 3- Applet2 is triggered 4- No exception is thrown. 18 Envelope Handler availability with			Applet1 finalizes	
Envelope Handler availability with EVENT_EVENT_DOWNLOAD_IDLE_SCREEN_ AVAILABLE 1-Envelope event download idle screen available is sent to the SIM 1- Applet1 is triggered 2-EnvelopeHandler.getTheHandler() method is called by Applet1 2- No exception is thrown. Applet1 finalizes 3-EnvelopeHandler.getTheHandler() method is called by Applet2 3- Applet2 is triggered 4- No exception is thrown.			3- Applet2 is triggered	
EVENT_EVENT_DOWNLOAD_IDLE_SCREEN_AVAILABLE 1-Envelope event download idle screen available is sent to the SIM 1- Applet1 is triggered 2-EnvelopeHandler.getTheHandler() method is called by Applet1 2- No exception is thrown. Applet1 finalizes 3-EnvelopeHandler.getTheHandler() method is called by Applet2 3- Applet2 is triggered 4- No exception is thrown.	47	Envelope Handler availability with	4- No exception is thrown	
available is sent to the SIM 2-EnvelopeHandler.getTheHandler() method is called by Applet1 2- No exception is thrown. 3-EnvelopeHandler.getTheHandler() method is called by Applet2 3- Applet1 finalizes 3- Applet2 is triggered 4- No exception is thrown.	1/	EVENT_EVENT_DOWNLOAD_IDLE_SCREEN_		
is called by Applet1 2- No exception is thrown. Applet1 finalizes 3-EnvelopeHandler.getTheHandler() method is called by Applet2 3- Applet2 is triggered 4- No exception is thrown.			1- Applet1 is triggered	
3-EnvelopeHandler.getTheHandler() method is called by Applet2 3- Applet2 is triggered 4- No exception is thrown.			2- No exception is thrown.	
is called by Applet2 3- Applet2 is triggered 4- No exception is thrown. 18 Envelope Handler availability with			Applet1 finalizes	
18 Envelope Handler availability with			3- Applet2 is triggered	
			4- No exception is thrown.	
EVENI_EVENI_DUWNLUAD_CAKD_KEADEK	18	Envelope Handler availability with EVENT_EVENT_DOWNLOAD_CARD_READER		

ld	Description	API/Framework Expectation	APDU Expectation
	_STATUS		
	1-Envelope event download card reader status is sent to the SIM	1- Applet1 is triggered	
	2-EnvelopeHandler.getTheHandler() method is called by Applet1	2- No exception is thrown.	
		Applet1 finalizes	
	3-EnvelopeHandler.getTheHandler() method is called by Applet2	3- Applet2 is triggered	
		4- No exception is thrown.	
19	Envelope Handler availability with EVENT_EVENT_DOWNLOAD_LANGUAGE_ SELECTION		
	1-Envelope event download language selection is sent to the SIM	1- Applet1 is triggered	
	2-EnvelopeHandler.getTheHandler() method is called by Applet1	2-No exception is thrown.	
	Applet1 finalizes.	Applet1 finalizes. Applet2 is triggered	
	3-EnvelopeHandler.getTheHandler() method is called by Applet2		
		3-No exception is thrown.	
20	Envelope Handler availability with EVENT_EVENT_DOWNLOAD_BROWSER_ TERMINATION		
	1-Envelope event download browser termination is sent to the SIM	1- Applet1 is triggered	
	2-EnvelopeHandler.getTheHandler() method is called by Applet1	2-No exception is thrown.	
		Applet1 finalizes. Applet2 is triggered	
	3-EnvelopeHandler.getTheHandler() method is called by Applet2	3-No exception is thrown.	
21	Envelope Handler availability with EVENT_STATUS_COMMAND		
	1-Status command is sent to the SIM	1- Applet1 is triggered	
	2-EnvelopeHandler.getTheHandler() method is called by Applet1	2- A Toolkit exception HANDLER_NOT_AVAILABLE is thrown	
		Applet1 finalizes.	
		3- Applet2 is triggered	
	3-EnvelopeHandler.getTheHandler() method is called by Applet2	4- A Toolkit exception HANDLER_NOT_AVAILABLE is thrown	
22	Envelope Handler availability with EVENT_EVENT_DOWNLOAD_DATA_AVAILAB LE		
	1- Applet1 builds a proactive command OPEN CHANNEL. proactiveHandler.send() method is called	1- Applet1 is registered to EVENT_EVENT_DOWNLOAD_DA TA_AVAILABLE and EVENT_EVENT_DOWNLOAD_CH	1- OPEN CHANNEL proactive command is fetched

ld	Description	API/Framework Expectation	APDU Expectation
		ANNEL_STATUS	TERMINAL RESPONSE is
	2-Envelope event download data available is sent to the SIM with ChannelId=01.	2- Applet1 is triggered	issued with Channel Id = 01
	3-EnvelopeHandler.getTheHandler() method is called by Applet1	3-No exception is thrown.	
23	Envelope Handler availability with EVENT_EVENT_DOWNLOAD_CHANNEL_STAT US		
	1-Envelope event download channel status is sent to the SIM with ChannelId=01.	1- Applet1 is triggered	
	2-EnvelopeHandler.getTheHandler() method is called by Applet1	2-No exception is thrown.	
24	Envelope Handler availability with EVENT_ UNRECOGNIZED_ENVELOPE		
	1-An unrecognized Envelope is sent to the SIM	1- Applet1 is triggered	
	2-EnvelopeHandler.getTheHandler() method is called by Applet1	2- No exception is thrown.	
		Applet1 finalizes Applet2 is triggered	
	3-EnvelopeHandler.getTheHandler() method is called by Applet2	3- No exception is thrown.	
25	Envelope Handler availability with EVENT_FORMATTED_SMS_PP_UPD		
	1- A formatted Update Record EFsms instruction is sent to the SIM	1- Applet1 is triggered	
	2-EnvelopeHandler.getTheHandler() method is called by Applet1	2- No exception is thrown.	
26	Envelope Handler availability with EVENT_UNFORMATTED_SMS_PP_UPD		
	1-An unformatted Update Record EFsms instruction is sent to the SIM	1- Applet1 is triggered	
	2-EnvelopeHandler.getTheHandler() method is called by Applet1	2- No exception is thrown.	
		Applet1 finalizes Applet2 is triggered	
	3-EnvelopeHandler.getTheHandler() method is called by Applet2	3- No exception is thrown.	

6.3.1.3.4 Test Coverage

CRR Number	Test Case Number
CRRN1	3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19,
	20, 22, 23, 24, 25, 26
CRRC1	1, 2, 21

6.3.1.4 EnvelopeResponseHandler

Test Area Reference: FWK_MHA_ERHD

6.3.1.4.1 Conformance Requirement

6.3.1.4.1.1 Normal Execution

• CRRN1: The handler is available for all triggered toolkit applets from the invocation of the processToolkit method of the toolkit applet until a toolkit applet has posted an envelope response or the first invocation of the ProactiveHandler.send method for the following events:

EVENT_FORMATTED_SMS_PP_ENV

EVENT_UNFORMATTED_SMS_PP_ENV

EVENT_CALL_CONTROL

EVENT_SMS_MO_CONTROL

EVENT_UNRECOGNIZED_ENVELOPE

- CRRN2: After a call to the post method the handler is not longer available.
- CRRN3: After a call to the send method the handler is not longer available.

6.3.1.4.1.2 Parameters error

No requirements.

6.3.1.4.1.3 Context Errors

• CRRC1: The handler is not available for the following events:

EVENT_FORMATTED_SMS_CB

EVENT_UNFORMATTED_SMS_CB

EVENT_MENU_SELECTION

EVENT_MENU_SELECTION_HELP_REQUEST

EVENT_TIMER_EXPIRATION

EVENT_EVENT_DOWNLOAD_MT_CALL

EVENT_EVENT_DOWNLOAD_CALL_CONNECTED

EVENT_EVENT_DOWNLOAD_CALL_DISCONNECTED

EVENT_EVENT_DOWNLOAD_LOCATION_STATUS

EVENT_EVENT_DOWNLOAD_USER_ACTIVITY

EVENT_EVENT_DOWNLOAD_IDLE_SCREEN_AVAILABLE

EVENT_EVENT_DOWNLOAD_CARD_READER_STATUS

EVENT_EVENT_DOWNLOAD_LANGUAGE_SELECTION

EVENT_EVENT_DOWNLOAD_BROWSER_TERMINATION

EVENT_STATUS_COMMAND

EVENT_PROFILE_DOWNLOAD

EVENT_FIRST_COMMAND_AFTER_SELECT

EVENT_EVENT_DOWNLOAD_DATA_AVAILABLE
EVENT_EVENT_DOWNLOAD_CHANNEL_STATUS
EVENT_FORMATTED_SMS_PP_UPD
EVENT_UNFORMATTED_SMS_PP_UPD

6.3.1.4.2 Test Suite Files

Test Script: FWK_MHA_ERHD_1.scr

Test Applet: FWK_MHA_ERHD_1.java

FWK_MHA_ERHD_2.java

Load Script: FWK_MHA_ERHD_1.ldr

Cleanup Script: FWK_MHA_ERHD_1.clr

Parameter File: FWK_MHA_ERHD_1.par

6.3.1.4.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1	Toolkit Applet1 and Toolkit Applet2 registration and Envelope Response Handler availability with EVENT_FIRST_COMMAND_AFTER_SELECT 1- Applet1 is registered to all events defined in TS 43.019 [7]. Using the methods initMenuEntry() for EVENT_MENU_SELECTION, requestPollInterval() for EVENT_STATUS_COMMAND, allocateTimer() for EVENT_TIMER_EXPIRATION and setEventList() for the rest of the events.	71 W Tullion of the Exposition	Al Bo Exposition
	Applet2 is registered to EVENT_UNFORMATTED_SMS_PP_ENV and EVENT_UNRECOGNIZED_ENVELOPE.	1- No exception is thrown	
	3- Select MF. 3-EnvelopeResponseHandler.getTheHandler() method is called by Applet1 Applet1 is deregistered to EVENT_FIRST_COMMAND_AFTER_SELECT.	2- Applet1 is triggered by EVENT_FIRST_COMMAND_AFTE R_SELECT	
		3- A Toolkit exception HANDLER_NOT_AVAILABLE is thrown	
2	Handler availability with EVENT_PROFILE_DOWNLOAD		
	1- Terminal Profile command is sent to the SIM without the facility of SET_EVENT_LIST, SETUP_IDLE_MODE_TEXT, SETUP_MENU and POLL_INTERVAL.	1- Applet1 Is Triggered By EVENT_PROFILE_DOWNLOAD	
	2- EnvelopeResponseHandler.getTheHandler() method is called by Applet1 Applet1 is deregistered to EVENT_PROFILE_DOWNLOAD	2- A Toolkit exception HANDLER_NOT_AVAILABLE is thrown	
3	Envelope Response Handler availability with EVENT_MENU_SELECTION_HELP_REQUEST		

ld	Description	API/Framework Expectation	APDU Expectation
	Perform SIM initialization with all the facilities supported		
	1-Envelope menu selection with help request is sent to the SIM	1- Applet1 is triggered.	
	2-EnvelopeResponseHandler.getTheHandler() method is called by Applet1		
		2- A Toolkit exception HANDLER_NOT_AVAILABLE is thrown	
4	Envelope Response Handler availability with EVENT_MENU_SELECTION		
	1-A envelope menu selection is sent to the SIM	1- Applet1 is triggered	
	2-EnvelopeResponseHandler.getTheHandler()		
	method is called by Applet1	2- A Toolkit exception HANDLER_NOT_AVAILABLE is thrown	
5	Envelope Response Handler availability with EVENT_FORMATTED_CB		
	1-Envelope cell broadcast formatted is sent to the SIM	1- The applet1 is triggered.	
	2-EnvelopeResponseHandler.getTheHandler() method is called by Applet1	2-A Toolkit exception HANDLER_NOT_AVAILABLE is thrown	
6	Envelope Response Handler availability with EVENT_UNFORMATTED_CB		
	1-Envelope cell broadcast unformatted is sent to the SIM	1- Applet1 is triggered.	
	2-EnvelopeResponseHandler.getTheHandler() method is called by Applet1	2- A Toolkit exception HANDLER_NOT_AVAILABLE is thrown	
7	Envelope Response Handler availability with EVENT_TIMER_EXPIRATION		
	1-Envelope Timer Expiration is sent to the SIM	1- Applet1 is triggered.	
	2-EnvelopeResponseHandler.getTheHandler() method is called by Applet1	2- A Toolkit exception HANDLER_NOT_AVAILABLE is thrown	
8	Envelope Response Handler availability with EVENT_EVENT_DOWNLOAD_MT_CALL		
	1-Envelope event download mt call is sent to the SIM	1- Applet1 is triggered.	
	2-EnvelopeResponseHandler.getTheHandler()	2 -A Toolkit exception	

ld	Description	API/Framework Expectation	APDU Expectation
	method is called by Applet1	HANDLER_NOT_AVAILABLE is	
		thrown	
9	Envelope Response Handler availability with		
	EVENT_EVENT_DOWNLOAD_CALL_CONNECT		
	ED		
	1 800001000 000000 10000100100000000000		
	1-Envelope event download call connected is sent to the SIM	1- Applet1 is triggered.	
	2-EnvelopeResponseHandler.getTheHandler()		
	method is called by Applet1	2- A Toolkit exception	
		HANDLER_NOT_AVAILABLE is thrown	
		unown	
10	Envelope Response Handler availability with		
	EVENT_EVENT_DOWNLOAD_CALL_DISCONN		
	ECTED		
	1-Envelope event download call	1- Applet1 is triggered.	
	disconnected is sent to the SIM	, Applet is aliggered.	
		O A Tablitana atian	
	2-EnvelopeResponseHandler.getTheHandler() method is called by Applet1	2- A Toolkit exception HANDLER_NOT_AVAILABLE is	
	method is carred by Appreti	thrown	
11	Envelope Response Handler availability with		
	EVENT_EVENT_DOWNLOAD_LOCATION_STA		
	103		
	1-Envelope event download location status	1- Applet1 is triggered.	
	is sent to the SIM		
	2-Applet1 obtains the Envelope Response		
	Handler	2- A Toolkit exception	
		HANDLER_NOT_AVAILABLE is	
		thrown	
12	Envelope Response Handler availability with		
	EVENT_EVENT_DOWNLOAD_USER_ACTIVITY		
	1 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7		
	1-Envelope event download user activity is sent to the SIM	1- Applet1 is triggered.	
	2-EnvelopeResponseHandler.getTheHandler()	2- A Toolkit exception	
	method is called by Applet1	HANDLER_NOT_AVAILABLE is	
		thrown	
13	Envelope Response Handler availability with		
	EVENT_EVENT_DOWNLOAD_IDLE_SCREEN_		
	AVAILABLE		
	1-Envelope event download idle screen	1- Applet1 is triggered.	
	available is sent to the SIM		
	_		
	2-EnvelopeResponseHandler.getTheHandler() method is called by Applet1	2- A Toolkit exception	
	meened is carred by Appreci	HANDLER_NOT_AVAILABLE is	
		thrown	

ld	Description	API/Framework Expectation	APDU Expectation
14	Envelope Response Handler availability with EVENT_EVENT_DOWNLOAD_CARD_READERSTATUS		
	1-Envelope event download card reader status is sent to the SIM	1- Applet1 is triggered	
	2-EnvelopeResponseHandler.getTheHandler() method is called by Applet1	2- A Toolkit exception HANDLER_NOT_AVAILABLE is thrown	
15	Envelope Response Handler availability with EVENT_EVENT_DOWNLOAD_LANGUAGE_ SELECTION		
	1-Envelope event download language selection is sent to the SIM	1- Applet1 is triggered	
	2-EnvelopeResponseHandler.getTheHandler() method is called by Applet1	2-A Toolkit exception HANDLER_NOT_AVAILABLE is thrown	

ld	Description	API/Framework Expectation	APDU Expectation
16	Envelope Response Handler availability with EVENT_EVENT_DOWNLOAD_BROWSER_ TERMINATION		
	1-Envelope event download browser termination is sent to the SIM	1- Applet1 is triggered	
	2-EnvelopeResponseHandler.getTheHandler() method is called by Applet1	2-A Toolkit exception HANDLER_NOT_AVAILABLE is thrown	
17	Envelope Response Handler availability with EVENT_STATUS_COMMAND		
	1-Status command is sent to the SIM	1- Applet1 is triggered	
	2-EnvelopeResponseHandler.getTheHandler() method is called by Applet1	2- A Toolkit exception HANDLER_NOT_AVAILABLE is thrown	
18	Envelope Response Handler availability with EVENT_EVENT_DOWNLOAD_DATA_AVAILAB LE		1- The OPEN CHANNEL command is fetched.
	1- Applet1 initialises a proactive command OPEN CHANNEL and calls the send() method.	2- Applet1 is triggered	TERMINAL RESPONSE IS SENT TO THE SIM with
	2- Envelope event download data avalaible is sent to the SIM with channelId=01	3- A Toolkit exception HANDLER_NOT_AVAILABLE is thrown	channelld=01
	3-EnvelopeResponseHandler.getTheHandler() method is called by Applet1		

ld	Description	API/Framework Expectation	APDU Expectation
19	Envelope Response Handler availability with EVENT_EVENT_DOWNLOAD_CHANNEL_STAT US		
		1- Applet1 is triggered	
		2- A Toolkit exception HANDLER_NOT_AVAILABLE is thrown	
	2-EnvelopeResponseHandler.getTheHandler() method is called by Applet1		

ld	Description	API/Framework Expectation	APDU Expectation
20	Envelope Response Handler availability with EVENT_FORMATTED_SMS_PP_UPD		
	1- Update Record EFsms instruction formatted is sent to the SIM	1- The applet1 is triggered.	
	2-EnvelopeResponseHandler.getTheHandler() method is called by Applet1	2-A Toolkit exception HANDLER_NOT_AVAILABLE is thrown	

ld	Description	API/Framework Expectation	APDU Expectation
21	Envelope Response Handler availability with EVENT_UNFORMATTED_SMS_PP_UPD		
	1- Update Record EFsms instruction unformatted is sent to the SIM	1- Applet1 is triggered.	
	2-EnvelopeResponseHandler.getTheHandler() method is called by Applet1	2- A Toolkit exception HANDLER_NOT_AVAILABLE is thrown	

ld	Description	API/Framework Expectation	APDU Expectation
22	Envelope Response Handler availability with EVENT_FORMATTED_SMS_PP_ENV		
	1-A formatted sms pp envelope is sent to the SIM	1- Applet1 is triggered	
	2-EnvelopeResponseHandler.getTheHandler() method is called by Applet1	2- No exception is thrown.	
	3-Applet1 builds an additional information for response packet and it calls the post method		3- The response packet is sent
	4-Applet1 calls all methods of the Envelope Response Handler (including the inherited method)	4- A Toolkit exception HANDLER_NOT_AVAILABLE is thrown for each method	
	5-A EVENT_FORMATTED_SMS_PP_ENV envelope is sent to the SIM	Applet1 finalizes 5- Applet1 is triggered	
	6-EnvelopeResponseHandler.getTheHandler() method is called by Applet1	6- No Exception is thrown	7. The proceeding command
	7-Applet1 builds a proactive command and it calls the send() method		7- The proactive command is sent
	8-Applet1 calls all methods of the Envelope Response Handler (including the inherited method)	8- Toolkit exception HANDLER_NOT_AVAILABLE is thrown for each method	

ld	Description	API/Framework Expectation	APDU Expectation
23	Envelope Response Handler availability with EVENT_UNFORMATTED_SMS_PP_ENV		
	1-An unformatted sms pp envelope is sent to the SIM	1- Applet1 is triggered	
	2-EnvelopeResponseHandler.getTheHandler() method is called by Applet1	2- No exception is thrown.	
	3-Applet1 builds the envelope response and it calls the post() method		3- The envelope response is
	4- Applet1 calls all methods of the Envelope Response Handler (including the inherited method)	4- A Toolkit exception HANDLER_NOT_AVAILABLE is thrown for each method Applet1 finalizes	sent
		5- Applet2 is triggered.	
	5-EnvelopeResponseHandler.getTheHandler() method is called	A Toolkit exception HANDLER_NOT_AVAILABLE is thrown.	
	6-An unformatted sms pp envelope is sent to the SIM	Applet2 finalizes	
	to the SIM	6- Applet1 is triggered.	
	7-EnvelopeResponseHandler.getTheHandler() method is called.	7. No expension is thrown	
	8-Applet1 builds a proactive command and it calls the send() method	7- No exception is thrown.	
	9-Applet1 calls all methods of the Envelope Response Handler (including the inherited method)		9- The proactive command is fetched and the Terminal response is issued.
	imerica menoa)	9- A Toolkit exception HANDLER_NOT_AVAILABLE is thrown for each method.	
		Applet1 finalizes	
	10-EnvelopeResponseHandler.getTheHandler() method is called by Applet2	10- Applet2 is triggered.	
		A Toolkit exception HANDLER_NOT_AVAILABLE is thrown.	

ld	Description	API/Framework Expectation	APDU Expectation
24	Envelope Response Handler availability with EVENT_CALL_CONTROL_BY_SIM		
	1-Envelope call control by sim is sent to the SIM	1- Applet1 is triggered	
	2-EnvelopeResponseHandler.getTheHandler() method is called by Applet1		
	3-Applet1 builds the envelope response and it calls the postAsBERTLV() method	2- No exception is thrown.	3- The envelope response is sent
	4-Applet1 calls all methods of the Envelope Response Handler (including the inherited method)	4- Toolkit exception	
	5-Envelope call control by sim is sent to the SIM	HANDLER_NOT_AVAILABLE is thrown for each method Applet1 finalizes	
	6-EnvelopeResponseHandler.getTheHandler() method is called by Applet1	5- Applet1 is triggered	
	7-Applet1 builds a proactive command and it calls the send() method	6- No Exception is thrown	
	8-Applet1 calls all methods of the Envelope Response Handler (including the inherited method)		7- The proactive command is fetched and the Terminal response is issued
		8- A Toolkit exception HANDLER_NOT_AVAILABLE is thrown for each method	

ld	Description	API/Framework Expectation	APDU Expectation
25	Envelope Response Handler availability with EVENT_MO_SHORT_MESSAGE_CONTROL_B Y_SIM		
	1-Envelope mo short message control by sim is sent to the SIM	1- Applet1 is triggered	
	2-EnvelopeResponseHandler.getTheHandler() method is called by Applet1	2- No exception is thrown.	
	3-Applet1 builds the envelope response and it calls the postAsBERTLV() method		3-The envelope response is sent
	4-Applet1 calls all methods of the Envelope Response Handler (including the inherited method)	4- A Toolkit exception HANDLER_NOT_AVAILABLE is thrown for each method	
	5-Envelope mo short message control by sim is sent to the SIM	Applet1 finalizes 5- Applet1 is triggered	
	6-EnvelopeResponseHandler.getTheHandler() method is called by Applet1	o y ppietrie anggeres	
	7-Applet1 builds a proactive command and it calls the send method	6- No exception is thrown	7- The proactive command
	8-Applet1 calls all methods of the Envelope Response Handler (including the inherited method)		is fetched and the Terminal Response is issued
		8- A Toolkit exception HANDLER_NOT_AVAILABLE is thrown for each method	

Description	API/Framework Expectation	APDU Expectation
Envelope Response Handler availability with EVENT_UNRECOGNIZED_ENVELOPE		
	1- Applet1 is triggered	
	2- No exception is thrown.	
		3- The envelope response is
Response Handler (including the inherited	4- A Toolkit exception HANDLER_NOT_AVAILABLE is thrown for each method	sent
	Applet1 finalizes 5- Applet2 is triggered.	
	A Toolkit exception HANDLER_NOT_AVAILABLE is	
	thrown.	
	Applet2 finalizes 6- Applet1 is triggered.	
Envelope Response Handler (including the	7- No exception is thrown.	9- The proactive command is fetched and the Terminal response is issued
	9- A Toolkit exception HANDLER_NOT_AVAILABLE is thrown for each method	
	Applet1 finalizes	
	10- Applet2 is triggered	
	A Toolkit exception HANDLER_NOT_AVAILABLE is thrown.	
	EVENT_UNRECOGNIZED_ENVELOPE 1-An unrecognized Envelope is sent to the SIM 2-EnvelopeResponseHandler.getTheHandler() method is called by Applet1 3-Applet1 builds the envelope response and	EVENT_UNRECOGNIZED_ENVELOPE 1-An unrecognized Envelope is sent to the SIM 2-EnvelopeResponseHandler.getTheHandler() method is called by Applet1 3-Applet1 builds the envelope response and it calls all methods of Envelope Response Handler (including the inherited method) 4-Applet1 calls all methods of Envelope Response Handler (including the inherited method is called 5-EnvelopeResponseHandler.getTheHandler() method is called 6-An unrecognized Envelope is sent to the SIM 7-EnvelopeResponseHandler.getTheHandler() method is called 8-Applet1 builds a proactive command and it calls the send() method 9-Applet1 calls all methods of the Envelope Response Handler (including the inherited method) 7- No exception is thrown. Applet2 finalizes 6- Applet1 is triggered. 7- No exception is thrown. 10-EnvelopeResponseHandler.getTheHandler() method is called by Applet2 10-Applet2 is triggered A Toolkit exception HANDLER_NOT_AVAILABLE is thrown for each method Applet1 finalizes 10-Applet2 is triggered A Toolkit exception HANDLER_NOT_AVAILABLE is thrown for each method Applet1 finalizes 10-Applet2 is triggered A Toolkit exception HANDLER_NOT_AVAILABLE is thrown for each method Applet1 finalizes

ld	Description	API/Framework Expectation	APDU Expectation
27	The envelope response is sent when a proactive session is ongoing 1-A formatted SMS PP envelope is sent to the SIM.	1- Applet1 is triggered.	,
	2-Proactive command DISPLAY TEXT is built and it calls the send() method. 3-A call control by sim envelope is sent to the SIM.	3- Applet1 is triggered	2- 91 XX
	4-EnvelopeResponseHandler.getTheHandler() method is called by Applet1 5-Applet1 builds the envelope response and it calls the postAsBERTLV	4- No exception is thrown	5 The envelope response is
			5-The envelope response is sent 9F YY GET RESPONSE Data 91 XX
			Fetch DISPLAY TEXT Terminal Response DISPLAY TEXT

ld	Description	API/Framework Expectation	APDU Expectation
28	Envelope Response Handler availability with EVENT_UNFORMATTED_SMS_PP_ENV in case of multi-triggering		
		1- Applet1 is triggered	
	1-A unformatted sms pp envelope is sent to the SIM	2- No exception is thrown.	
	2-EnvelopeResponseHandler.getTheHandler() method is called by Applet1	3- Applet1 finalizes	
		4- Applet2 is triggered.	
	5-EnvelopeResponseHandler.getTheHandler() method is called by Applet 2	5- No Exception is thrown	6. The response is checked.
	6- Applet2 calls the post() method		
		Applet2 finalizes	

ld	Description	API/Framework Expectation	APDU Expectation
29	Envelope Response Handler availability with EVENT_UNRECOGNIZED_ENVELOPE in case of multi-triggering	1- Applet1 is triggered	
		2- No exception is thrown.	
	1-An unrecognized Envelope is sent to the SIM		
	2-EnvelopeResponseHandler.getTheHandler() method is called by Applet1	3- Applet1 finalizes	
		4- Applet2 is triggered.	
		5- No Exception is thrown	
	5-EnvelopeResponseHandler.getTheHandler() method is called by Applet 2	Applet2 finalizes	6- The response is checked
	6- Applet2 calls the post() method	Applete mailes	o- The response is checked

6.3.1.4.4 Test Coverage

CRR Number	Test Case Number
CRRN1	20, 21, 22, 23, 24, 25,26,27
CRRN2	20, 21, 22, 23, 24, 25
CRRN3	20, 21, 22, 23, 24, 25
CRRC1	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 28, 29
	14, 15, 16, 17, 18, 19, 28, 29

6.3.2 Handler Integrity

6.3.2.1 ProactiveHandler

Test Area Reference: FWK_HIN_PAHD

6.3.2.1.1 Conformance Requirement

6.3.2.1.1.1 Normal Execution

• CRRN1: At the processToolkit invocation the TLV-List is cleared.

• CRRN2: After a call to ProactiveHandler.send method the handler will remain unchanged until the ProactiveHandler.init or appendTLV method are called.

6.3.2.1.1.2 Parameters error

No requirements.

6.3.2.1.1.3 Context Errors

No requirements.

6.3.2.1.2 Test Suite Files:

Test Script: FWK_HIN_PAHD_1.scr

Test Applet: FWK_HIN_PAHD_1.java

FWK_HIN_PAHD_2.java

Load Script: FWK_HIN_PAHD_1.ldr

Cleanup Script: FWK_HIN_PAHD_1.clr

Parameter File: FWK_HIN_PAHD_1.par

6.3.2.1.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1	At the processToolkit invocation the TLV-List		-
	is cleared		
	Applet1 and Applet2 are registered to		
	EVENT_UNFORMATTED_SMS_PP_ENV.		
	1-An envelope containing an unformatted	1- Applet1 is triggered.	
	sms pp is sent to the SIM		
	2-ProactiveHandler.getLength() method is	2- The return value is 0	
	called by Applet1		
2	TLV-List change after the init method		
	invocation		
	ProactiveHandler.init() method is called		
	by Applet1		
	1-ProactiveHandler.getLength() method is		
	called by Applet1	1- The return value is 9	
3	The TLV-List remains unchanged after the		
	send() method invocation		
	1-ProactiveHandler.send() method is called		1- The proactive command
	by Applet1		is fetched and the terminal
			response is issued.
	2-ProactiveHandler.getLength() method is	2- The return value is 9, and its	
	called by Applet1	contents is the same than before	
		the calling to send method	
	It's checked that the content is the same		
	than before the calling to send method		
	using ProactiveHandler.copyValue and Util.arrayCompare methods		
	otil.allaycompale medioab		
4	At the processToolkit invocation the TLV-List		
	is cleared		
		1- Applet2 is triggered	
	1-ProactiveHandler.getLength() method is	2. The meturn value is 0	
	called by Applet2	2- The return value is 0	
	2 DroagtivoHandler gotValuatenath()		
	<pre>2-ProactiveHandler.getValueLength() method is called by Applet2</pre>	3- ToolkitException	
		UNAVAILABLE_ELEMENT is	
		thrown	

6.3.2.1.4 Test Coverage

CRR Number	Test Case Number	
CRRN1	1, 2, 3, 4	
CRRN2	3	

6.3.2.2 ProactiveResponseHandler

Test Area Reference: FWK_HIN_ PRHD

6.3.2.2.1 Conformance Requirement

6.3.2.2.1.1 Normal Execution

- CRRN1: The ProactiveResponseHandler content is changed after the call to ProactiveHandler.send method and remains unchanged until next call to the ProactiveHandler.send method.
- CRRN2: The ProactiveResponseHandler may not be available before the first call to ProactiveHandler.send method, if available the content is cleared.

6.3.2.2.1.2 Parameters error

No requirements.

6.3.2.2.1.3 Context Errors

No requirements.

6.3.2.2.2 Test Suite Files

Test Script: FWK_HIN_PRHD_1.scr

Test Applet: FWK_HIN_PRHD_1.java

Load Script: FWK_HIN_PRHD_1.ldr

Cleanup Script: FWK_HIN_PRHD_1.clr

Parameter File: FWK_HIN_PRHD_1.par

6.3.2.2.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1	Applet registration and		
	ProactiveResponseHandler obtaining		
	1-Applet is registered to all events defined in [7].		
	Using the methods initMenuEntry for		
	EVENT_MENU_SELECTION, requestPollInterval() for		
	EVENT_STATUS_COMMAND, allocateTimer() for		
	EVENT_TIMER_EXPIRATION and setEventList() for the rest of the events.	1- No exception is thrown	
	Terminal Profile command is sent to the SIM without the facilities of		
	SET_EVENT_LIST ,SETUP_IDLE_MODE_TEXT,		
	SETUP_MENU and POLL_INTERVAL.	2- Applet is triggered.	
	For each event:	3- Behaviour 1:	
	2-ProactiveResponseHandler.getTheHandler()	Toolkit Exception HANDLER_NOT_AVAILABLE is	
	is called	thrown.	
		Behaviour 2:	
	If handler is available,	No exception is thrown, the return value is 0	
	ProactiveResponseHandler.getLength() is	value is 0	
	called		
2	The ProactiveResponseHandler remains		
	unchanged after send method invocation until next send method invocation		
	1-Applet builds a proactive command	1. The Dreading Deepended Landler	2 A propostive command is
	ProactiveHandler.send() method is called	1- The ProactiveResponseHandler contains the terminal response	2- A proactive command is fetched
			The terminal response is sent with length 12
	2-ProactiveResponseHandler.getLength() method is called	3- The return value is 12	
	3-ProactiveHandler.init() method is called	4- No exception is thrown and the Proactive Response Handler remains unchanged	
		3	
	4-ProactiveHandler.send() method is called	5- The ProactiveResponseHandler	6. A proportive command is
		contains the terminal response of the second proactive command	6- A proactive command is fetched The terminal response is sent with length 15
		7. The mature value := 45	Ŭ
	5-ProactiveResponseHandler.getLength() method is called	7- The return value is 15	

6.3.2.2.4 Test Coverage

CRR Number	Test Case Number	
CRRN1	1, 2	
CRRN2	1	

6.3.2.3 EnvelopeHandler

Test Area Reference: FWK_HIN_ENHD

6.3.2.3.1 Conformance Requirement

6.3.2.3.1.1 Normal Execution

- CRRN1: The EnvelopeHandler and its content are available for all triggered toolkit applets, from the invocation to the termination of their processToolkit method
- CRRN2: The SIM Toolkit Framework guarantees that all triggered toolkit applets receive the data.
- CRRN3: The SIM Toolkit Framework shall convert the Update Record EFsms in the EnvelopeHandler TLV List containing Device Identities TLV, Address TLV and SMS TPDU TLV.
- CRRN4: The getEnvelopeTag() method shall return BTAG_SMS_PP_DOWNLOAD.
- CRRN5: The getLength() method shall return the Simple TLV list length.
- CRRN6 The Device Identity Simple TLV is used to store the information about the absolute record number in the EFsms file and the value of the EFsms record status byte.

6.3.2.3.1.2 Parameters error

No requirements.

6.3.2.3.1.3 Context Errors

No requirements.

6.3.2.3.2 Test Suite Files

Test Script: FWK_HIN_ENHD_1.scr

Test Applet: FWK_HIN_ENHD_1.java

Load Script: FWK_HIN_ENHD_1.ldr

Cleanup Script: FWK_HIN_ENHD_1.clr

Parameter File: FWK_HIN_ENHD_1.par

6.3.2.3.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1	Applet initialization and Envelope Handler	,	
	integrity checks with		
	EVENT_MENU_SELECTION_HELP_REQUEST		
	1- Applet is registered to all events defined in TS 43.019 [7] except EVENT_PROFILE_DOWNLOAD and EVENT_STATUS_COMMAND. Using the methods initMenuEntry() for	1-No exception is thrown	
	<pre>EVENT_MENU_SELECTION, allocateTimer()for EVENT_TIMER_EXPIRATION, and setEventList() for the rest of the events.</pre>		
	Perform SIM initialization with all the facilities supported		
	2-Envelope menu selection with help request is sent to the SIM	2- Applet is triggered3- No exception is thrown.	
	3-EnvelopeHandler.getTheHandler() method is called	'	
	4-Copy the contents of the envelope handler in buffer 1 using EnvelopeHandler.copy()	4- No exception is thrown	
	The EnvelopeHandler.findTLV() method is called with TAG_HELP_REQUEST		
	5-A proactive command DISPLAY TEXT is sent		5- 91 xx.
	6-Envelope call control by sim is sent to SIM	6- Applet is triggered	
	EnvelopeHandler.getTheHandler() method is called		
	7- It's checked that the contents of the envelope handler is the envelope call control using EnvelopeHandler.copy() and Util.arrayCompare() methods	7- No exception is thrown and the handler contains the envelope call control by SIM	
	The EnvelopeHandler.findTLV() method is called with TAG_DEVICE_IDENTITIES		
	Call Control execution is finished.		
			A proactive command Display Text is fetched
			The terminal Response of DISPLAY TEXT is sent to the SIM
	Check that the TAG_HELP_REQUEST is the TLV selected		
	8-The contents of EnvelopeHandler are compared with buffer1 using Util.arrayCompare()	8- The contents of the envelope handler shall be the same as stored in buffer 1	
2	Envelope Handler integrity checks with EVENT_MENU_SELECTION		
	1-An envelope menu selection is sent to SIM	1- Applet is triggered	
	2-EnvelopeHandler.getTheHandler() method is called	2- No exception is thrown.	

ld	Description	API/Framework Expectation	APDU Expectation
	3-Copy the contents of the envelope handler in buffer 1 using EnvelopeHandler.copy() The EnvelopeHandler.findTLV() method is called with TAG_ITEM_IDENTIFIER	3- No exception is thrown.	
	4-A proactive command DISPLAY TEXT is sent		
	5-Envelope call control by sim is sent to SIM		4- 91 XX
	EnvelopeHandler.getTheHandler() method is called	5- Applet is triggered	
	6- It's checked the contents of the envelope handler is the envelope call control using EnvelopeHandler.copy() and Util.arrayCompare() methods	6- No exception is thrown and the handler contains the envelope call control by SIM	
	The EnvelopeHandler.findTLV() method is called with TAG_DEVICE_IDENTITIES	·	
	Call Control execution is finished.		
			Proactive command Display Text is fetched The terminal Response of DISPLAY TEXT is sent to
	It's checked that the TAG_ITEM_IDENTIFIER is the TLV selected		the SIM
	7- The contents of EnvelopeHandler are compared with bufferl using Util.arrayCompare()		
		7- The contents of the envelope handler shall be the same as stored in buffer 1	

ld	Description	API/Framework Expectation	APDU Expectation
3	Envelope Handler integrity checks with EVENT_FORMATTED_SMS_PP_ENV		
	1-A formatted sms pp envelope is sent to SIM	1- Applet is triggered	
	2-EnvelopeHandler.getTheHandler() method is called	2- No exception is thrown.	
	3-Copy the contents of the envelope handler in buffer 1 using EnvelopeHandler.copy()	3- No exception is thrown.	
	The EnvelopeHandler.findTLV() method is called with TAG_SMS_TPDU		
	4-A proactive command DISPLAY TEXT is sent		4- 91 XX
	5-Envelope call control by sim is sent to SIM	5- Applet is triggered	
	EnvelopeHandler.getTheHandler() method is called		
	6-It's checked that the contents of the envelope handler is the envelope call control using EnvelopeHandler.copy() and Util.arrayCompare methods	6- No exception is thrown and the handler contains the envelope call control by SIM	
	The EnvelopeHandler.findTLV() method is called with TAG_DEVICE_IDENTITIES		
	Call Control execution is finished.		
			Proactive command Display Text is fetched The terminal Response of DISPLAY TEXT is sent to the SIM
	It's checked that the TAG_SMS_TPDU is the TLV selected		-
	7- The contents of EnvelopeHandler are compared with buffer1 using Util.arrayCompare()	7- The contents of the envelope handler shall be the same as stored in buffer 1	

ld	Description	API/Framework Expectation	APDU Expectation
4	Envelope Handler integrity checks with EVENT_UNFORMATTED_SMS_PP_ENV		
	1-A unformatted sms pp envelope is sent to SIM	1- Applet is triggered	
	2-EnvelopeHandler.getTheHandler() method is called	2- No exception is thrown.	
	3-Copy the contents of the envelope handler in buffer 1 using EnvelopeHandler.copy()	3- No exception is thrown.	
	The EnvelopeHandler.findTLV method is called with TAG_DEVICE_IDENTITIES		4- 91 XX
	4-A proactive command DISPLAY TEXT is sent		
	5-Envelope call control by sim is sent to SIM	5- Applet is triggered	
	EnvelopeHandler.getTheHandler() method is called		
	6-It's checked that the contents of the envelope handler is the envelope call control using EnvelopeHandler.copy() and Util.arrayCompare() methods	6- No exception is thrown and the handler contains the envelope call control by SIM	
	The EnvelopeHandler.findTLV() method is called with TAG_DEVICE_IDENTITIES		
	Call Control execution is finished.		Proactive command Display Text is fetched
			The terminal Response of DISPLAY TEXT is sent to the SIM
	It's checked that the TAG_DEVICE_IDENTITIES is the TLV selected		
	7- The contents of EnvelopeHandler are compared with buffer1 using Util.arrayCompare()	7- The contents of the envelope handler shall be the same as stored in buffer 1.	

ld	Description	API/Framework Expectation	APDU Expectation
5	Envelope Handler integrity checks with	7.1. 91 Tallion of R. Expositation	7 ii 20 Expositation
	EVENT_UNFORMATTED_SMS_CB		
	1-A unformatted cellbroadcast envelope is sent to SIM	1- Applet is triggered	
	2-EnvelopeHandler.getTheHandler() method is called	2- No exception is thrown.	
	3-Copy the contents of the envelope handler in buffer 1 using EnvelopeHandler.copy()	3- No exception is thrown.	
	The EnvelopeHandler.findTLV() method is called with TAG_CELLBROADCAST_PAGE		4- 91 XX
	4-A proactive command DISPLAY TEXT is sent		
	5-Envelope call control by sim is sent to SIM	5- Applet is triggered	
	EnvelopeHandler.getTheHandler() method is		
	called	C. No avecantion is thrown and the	
	6-It's checked that the contents of the envelope handler is the envelope call control using EnvelopeHandler.copy and Util.arrayCompare() methods	6- No exception is thrown and the handler contains the envelope call control by SIM	
	The EnvelopeHandler.findTLV() method is called with TAG_DEVICE_IDENTITIES		
	Call Control execution is finished.		
			Proactive command Display Text is fetched
			The terminal Response of DISPLAY TEXT is sent to the SIM
	It's checked that the TAG_CELLBROADCAST_PAGE is the TLV selected	7- The contents of the envelope	
	7- The contents of EnvelopeHandler are compared with buffer1 using Util.arrayCompare()	handler shall be the same as stored in buffer 1.	
6	Envelope Handler integrity checks with		
	EVENT_TIMER_EXPIRATION		
	1-A timer expiration envelope is sent to SIM	1- Applet is triggered	
	2-EnvelopeHandler.getTheHandler() method is called	2- No exception is thrown.	
	3-Copy the contents of the envelope handler in buffer 1 using EnvelopeHandler.copy()	3- No exception is thrown.	
	The EnvelopeHandler.findTLV() method is called with TAG_TIMER_ID		
	4-A proactive command DISPLAY TEXT is sent		4- 91 XX
	5-Envelope call control by sim is sent to SIM	5- Applet is triggered	

ld	Description	API/Framework Expectation	APDU Expectation
	EnvelopeHandler.getTheHandler() method is called		
	6-It's checked that the contents of the envelope handler is the envelope call control using EnvelopeHandler.copy() and Util.arrayCompare() methods	6- No exception is thrown and the handler contains the envelope call control by SIM	
	The EnvelopeHandler.findTLV() method is called with TAG_DEVICE_IDENTITIES		
	Call Control execution is finished.		
			Proactive command Display Text is fetched
			The terminal Response of DISPLAY TEXT is sent to the SIM
	It's checked that the TAG_TIMER_ID is the TLV selected		
	7- The contents of EnvelopeHandler are compared with buffer1 using Util.arrayCompare()	7- The contents of the envelope handler shall be the same as stored in buffer 1	

ld	Description	API/Framework Expectation	APDU Expectation
7	Envelope Handler integrity checks with	,	
	EVENT_CALL_CONTROL_BY_SIM		
	1-A call control envelope is sent to SIM	1- Applet is triggered	
	2-EnvelopeHandler.getTheHandler() method is	2- No exception is thrown.	
	called	The exception is time with	
	3-Copy the contents of the envelope		
	handler in buffer 1 using EnvelopeHandler.copy()	3- No exception is thrown.	
	Enverspendidier.eopy()		
	The EnvelopeHandler.findTLV() method is		
	called with TAG_ADDRESS		
	4-A proactive command DISPLAY TEXT is sent		4- 91 XX
	5-Envelope call control by sim is sent to	5- Applet is triggered	
	SIM		
	EnvelopeHandler.getTheHandler() method is		
	called		
	6-It's checked that the contents of the	6- No exception is thrown and the	
	envelope handler is the envelope call	handler contains the envelope call	
	control using EnvelopeHandler.copy() and	control by SIM	
	Util.arrayCompare() methods	,	
	The EnvelopeHandler.findTLV() method is		
	called with TAG_DEVICE_IDENTITIES		
	Call Control execution is finished.		Danasti is a same and Disales.
			Proactive command Display Text is fetched
			l ext is retched
			The terminal Response of
			DISPLAY TEXT is sent to
			the SIM
	It's checked that the TAG_ADDRESS is the TLV selected	· · · · · ·	
	THV Beleeved	7- The contents of the envelope	
	7- The contents of EnvelopeHandler are	handler shall be the same as stored in buffer 1	
	compared with buffer1 using		
8	Util.arrayCompare() Envelope Handler integrity checks with EVENT_		
	MO_SHORT_MESSAGE_CONTROL_BY_SIM		
	1-A mo short message control by sim	1- Applet is triggered	
	envelope is sent to SIM		
		2. No expension is through	
	2-EnvelopeHandler.getTheHandler() method is	z- No exception is thrown.	
	called		
	3-Copy the contents of the envelope	3- No exception is thrown.	
	handler in buffer 1 using EnvelopeHandler.copy()		
	The EnvelopeHandler.findTLV() method is		
	called with TAG_ADDRESS		4- 91 XX
	4-A proactive command DISPLAY TEXT is sent		. 31700
	5-Envelope call control by sim is sent to SIM	5- Applet is triggered	
	DIT!		
	EnvelopeHandler.getTheHandler() method is		
	called		

ld	Description	API/Framework Expectation	APDU Expectation
	6-It's checked that the contents of the envelope handler is the envelope call control using EnvelopeHandler.copy() and Util.arrayCompare() methods	6- No exception is thrown and the handler contains the envelope call control by SIM	
	The EnvelopeHandler.findTLV() method is called with TAG_DEVICE_IDENTITIES		
	Call Control execution is finished.		Proactive command Display
			Text is fetched
	This shocked that the TAC ADDRESS is the		The terminal Response of DISPLAY TEXT is sent to the SIM
	It's checked that the TAG_ADDRESS is the TLV selected	7. The contents of the envelope	
	7- The contents of EnvelopeHandler are compared with buffer1 using Util.arrayCompare()	7- The contents of the envelope handler shall be the same as stored in buffer 1.	
9	Envelope Handler integrity checks with EVENT_ EVENT_DOWNLOAD_MT_CALL		
	1-A event download mt call envelope is sent to SIM	1- Applet is triggered	
	2-EnvelopeHandler.getTheHandler() method is called	2- No exception is thrown.	
	3-Copy the contents of the envelope handler in buffer 1 using EnvelopeHandler.copy()	3- No exception is thrown.	
	The EnvelopeHandler.findTLV() method is called with TAG_ADDRESS		
	4-A proactive command DISPLAY TEXT is sent		4- 91 XX
	5-Envelope call control by sim is sent to SIM	5- Applet is triggered	
	EnvelopeHandler.getTheHandler() method is called		
	6-It's checked that the contents of the envelope handler is the envelope call control using EnvelopeHandler.copy() and Util.arrayCompare() methods	6- No exception is thrown and the handler contains the envelope call control by SIM	
	The EnvelopeHandler.findTLV() method is called with TAG_DEVICE_IDENTITIES		
	Call Control execution is finished.		
			Proactive command Display Text is fetched
			The terminal Response of DISPLAY TEXT is sent to
	It's checked that the TAG_ADDRESS is the TLV selected		the SIM
	7- The contents of EnvelopeHandler are compared with buffer1 using Util.arrayCompare()	7- The contents of the envelope handler shall be the same as stored in buffer 1	

ld	Description	API/Framework Expectation	APDU Expectation
10	Envelope Handler integrity checks with EVENT_ EVENT_DOWNLOAD_CALL_CONNECTED	,	
	1-A event download call connected envelope is sent to SIM	1- Applet is triggered	
	2-EnvelopeHandler.getTheHandler() method is called	2- No exception is thrown.	
	3-Copy the contents of the envelope handler in buffer 1 using EnvelopeHandler.copy()	3- No exception is thrown.	
	The EnvelopeHandler.findTLV() method is called with TAG_ADDRESS		4 04 VV
	4-A proactive command DISPLAY TEXT is sent		4- 91 XX
	5-Envelope call control by sim is sent to SIM	5- Applet is triggered	
	EnvelopeHandler.getTheHandler() method is called		
	6-It's checked that the contents of the envelope handler is the envelope call control using EnvelopeHandler.copy and Util.arrayCompare() methods	6- No exception is thrown and the handler contains the envelope call control by SIM	
	The EnvelopeHandler.findTLV() method is called with TAG_DEVICE_IDENTITIES		
	Call Control execution is finished.		
			Proactive command Display Text is fetched
			The terminal Response of DISPLAY TEXT is sent to the SIM
	It's checked that the TAG_ADDRESS is the TLV selected	7- The contents of the envelope	
	7- The contents of EnvelopeHandler are compared with buffer1 using Util.arrayCompare()	handler shall be the same as stored in buffer 1.	

ld	Description	API/Framework Expectation	APDU Expectation
11	Envelope Handler integrity checks with EVENT_		
	EVENT_DOWNLOAD_CALL_DISCONNECTED		
	1-A event download call disconnected envelope is sent to SIM	1- Applet is triggered	
	envelope is sent to sim		
	2-EnvelopeHandler.getTheHandler() method is	2- No exception is thrown.	
	called	140 exception is timown.	
	3-Copy the contents of the envelope		
	handler in buffer 1 using	3- No exception is thrown.	
	EnvelopeHandler.copy()		
	The EnvelopeHandler.findTLV() method is		
	called with TAG_ADDRESS		
	1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2		4 04 VV
	4-A proactive command DISPLAY TEXT is sent		4- 91 XX
	5-Envelope call control by sim is sent to	5- Applet is triggered	
	SIM	19- Applet is triggered	
	EnvelopeHandler.getTheHandler() method is		
	called		
	6-It's checked that the contents of the	6- No exception is thrown and the	
	envelope handler is the envelope call control using EnvelopeHandler.copy and	handler contains the envelope call	
	Util.arrayCompare() methods	control by SIM	
	otilialia, compare() methods		
	The EnvelopeHandler.findTLV() method is		
	called with TAG_DEVICE_IDENTITIES		
	Call Control execution is finished.		
	call conclot execution is limiblea.		
			Proactive command Display
			Text is fetched
			The terminal Response of
			DISPLAY TEXT is sent to
			the SIM
	T. ()]]]		
	It's checked that the TAG_ADDRESS is the TLV selected		
	111 Bereded	7. The contents of the envelope	
	7- The contents of EnvelopeHandler are	7- The contents of the envelope handler shall be the same as stored	
	compared with buffer1 using	in buffer 1.	
12	Util.arrayCompare() Envelope Handler integrity checks with EVENT	in building.	
12	EVENT_DOWNLOAD_LOCATION_STATUS		
	1-A event download location status envelope	1- Applet is triggered	
	is sent to SIM		
1			
	2-EnvelopeHandler.getTheHandler() method is	2- No exception is thrown.	
	called		
	3-Copy the contents of the envelope handler	3- No exception is thrown	
	in buffer 1 using EnvelopeHandler.copy()		
	The EnvelopeHandler.findTLV() method is		
	called with TAG_LOCATION_STATUS		
	4-A proactive command DISPLAY TEXT is sent		4-91 XX
	5-Envelope call control by sim is sent to		
	GTM	5. Applot is triggered	
		5- Applet is triggered	
	EnvolopoHandlor cottheHandlor()		
	EnvelopeHandler.getTheHandler() method is called		
	041104	l	<u> </u>

ld	Description	API/Framework Expectation	APDU Expectation
	6-It's checked that the contents of the envelope handler is the envelope call control using EnvelopeHandler.copy() and Util.arrayCompare() methods	6- No exception is thrown and the handler contains the envelope call control by SIM	
	The EnvelopeHandler.findTLV() method is called with TAG_DEVICE_IDENTITIES		
	Call Control execution is finished.		
			Proactive command Display Text is fetched
			The terminal Response of DISPLAY TEXT is sent to the SIM
	It's checked that the TAG_LOCATION_STATUS is the TLV selected		and Onvi
	7- The contents of EnvelopeHandler are compared with buffer1 using Util.arrayCompare()	7- The contents of the envelope handler shall be the same as stored in buffer 1	
13	Envelope Handler integrity checks with EVENT_ EVENT_DOWNLOAD_USER_ACTIVITY		
	1-A event download user activity envelope is sent to SIM	1- Applet is triggered	
	2-EnvelopeHandler.getTheHandler() method is called	2- No exception is thrown.	
	3-Copy the contents of the envelope handler in buffer 1 using EnvelopeHandler.copy()	3- No exception is thrown.	
	The EnvelopeHandler.findTLV() method is called with TAG_DEVICE_IDENTITIES		
	4-A proactive command DISPLAY TEXT is sent		4- 91 XX
	5-Envelope call control by sim is sent to SIM	5- Applet is triggered	
	EnvelopeHandler.getTheHandler() method is called		
	6-It's checked that the contents of the envelope handler is the envelope call control using EnvelopeHandler.copy() and Util.arrayCompare() methods	6- No exception is thrown and the handler contains the envelope call control by SIM	
	The EnvelopeHandler.findTLV() method is called with TAG_DEVICE_IDENTITIES	,	
	Call Control execution is finished.		Proactive command Display Text is fetched
			The terminal Response of DISPLAY TEXT is sent to the SIM
	It's checked that the TAG_DEVICE_IDENTITIES is the TLV selected		
	7- The contents of EnvelopeHandler are compared with buffer1 using Util.arrayCompare()	7- The contents of the envelope handler shall be the same as stored in buffer 1	

ld	Description	API/Framework Expectation	APDU Expectation
14	Envelope Handler integrity checks with EVENT_ EVENT_DOWNLOAD_IDLE_SCREEN_AVAILAB LE		
	1-A event download idle screen available envelope is sent to SIM	1- Applet is triggered	
	2-EnvelopeHandler.getTheHandler() method is called	2- No exception is thrown.	
	3-Copy the contents of the envelope handler in buffer 1 using EnvelopeHandler.copy()	3- No exception is thrown.	
	The EnvelopeHandler.findTLV() method is called with TAG_DEVICE_IDENTITIES		
	4-A proactive command DISPLAY TEXT is sent		
	5-Envelope call control by sim is sent to SIM	5- Applet is triggered	4- 91 XX
	EnvelopeHandler.getTheHandler() method is called		
	6-It's checked that the contents of the envelope handler is the envelope call control using EnvelopeHandler.copy() and Util.arrayCompare() methods	6- No exception is thrown and the handler contains the envelope call control by SIM	
	The EnvelopeHandler.findTLV() method is called with TAG_DEVICE_IDENTITIES		
	Call Control execution is finished.		
			Proactive command Display Text is fetched
			The terminal Response of DISPLAY TEXT is sent to the SIM
	It's checked that the TAG_DEVICE_IDENTITIES is the TLV selected		and Onvi
	7- The contents of EnvelopeHandler are compared with buffer1 using Util.arrayCompare()	7- The contents of the envelope handler shall be the same as stored in buffer 1	

ld	Description	API/Framework Expectation	APDU Expectation
15	Envelope Handler integrity checks with EVENT_ EVENT_DOWNLOAD_CARD_READER_STATUS		
	1-A event download card reader status envelope is sent to SIM	1- Applet is triggered	
	2-EnvelopeHandler.getTheHandler() method is called	2- No exception is thrown.	
	3-Copy the contents of the envelope handler in buffer 1 using EnvelopeHandler.copy()	3- No exception is thrown.	
	The EnvelopeHandler.findTLV() method is called with TAG_CARD_READER_STATUS		
	4-A proactive command DISPLAY TEXT is sent		
	5-Envelope call control by sim is sent to SIM	5- Applet is triggered	4- 91 XX
	EnvelopeHandler.getTheHandler() method is called		
	6-It's checked that the contents of the envelope handler is the envelope call control using EnvelopeHandler.copy and Util.arrayCompare() methods	6- No exception is thrown and the	
	The EnvelopeHandler.findTLV() method is called with TAG_DEVICE_IDENTITIES	handler contains the envelope call control by SIM	
			Proactive command Display Text is fetched
	It's checked that the TAG_CARD_READER_STATUS is the TLV selected		The terminal Response of DISPLAY TEXT is sent to the SIM
	7- The contents of EnvelopeHandler are compared with buffer1 using Util.arrayCompare()	7- The contents of the envelope handler shall be the same as stored in buffer 1	

ld	Description	API/Framework Expectation	APDU Expectation
16	Envelope Handler integrity checks with UNRECOGNIZED_ENVELOPE		
	1-A unrecognized envelope is sent to SIM	1- Applet is triggered	
	2-EnvelopeHandler.getTheHandler() method is called	2- No exception is thrown.	
	3-Copy the contents of the envelope handler in buffer 1 using EnvelopeHandler.copy()	3- No exception is thrown.	
	4-A proactive command DISPLAY TEXT is sent		
	5-Envelope call control by sim is sent to SIM	.	4- 91 XX
		5- Applet is triggered	
	EnvelopeHandler.getTheHandler() method is called The EnvelopeHandler.getValueLength() is called		
	6-It's checked that the contents of the envelope handler is the envelope call control using EnvelopeHandler.copy() and Util.arrayCompare() methods	6- No exception is thrown and the handler contains the envelope call control by SIM	
	The EnvelopeHandler.findTLV() method is called with TAG_DEVICE_IDENTITIES		
	Call Control execution is finished.		
			Proactive command Display Text is fetched
			The terminal Response of DISPLAY TEXT is sent to the SIM
	7- The contents of EnvelopeHandler are compared with bufferl using Util.arrayCompare()	7- The contents of the envelope handler shall be the same as stored in buffer 1	

ld	Description	API/Framework Expectation	APDU Expectation
17	Envelope Handler integrity checks with EVENT_EVENT_DOWNLOAD_LANGUAGE_SEL ECTION		
	1-A event download language selection envelope is sent to SIM	1- Applet is triggered	
	2-EnvelopeHandler.getTheHandler() method is called	2-No exception is thrown.	
	3-Copy the contents of the envelope handler in buffer 1 using EnvelopeHandler.copy()	3-No exception is thrown.	
	The EnvelopeHandler.findTLV() method is called with TAG_EVENT_LIST 4-A proactive command DISPLAY TEXT is sent		
	5-Envelope call control by sim is sent to SIM	5- Applet is triggered	4-91 XX
	EnvelopeHandler.getTheHandler() method is called		
	6-It's checked that the contents of the envelope handler is the envelope call control using EnvelopeHandler.copy() and Util.arrayCompare() methods	6- No exception is thrown and the handler contains the envelope call control by SIM	
	The EnvelopeHandler.findTLV() method is called with TAG_DEVICE_IDENTITIES		
	Call Control execution is finished.		
			Proactive command Display Text is fetched
			The terminal Response of DISPLAY TEXT is sent to the SIM
	It's checked that the TAG_EVENT_LIST is the TLV selected		
	7- The contents of EnvelopeHandler are compared with buffer1 using Util.arrayCompare()	7- The contents of the envelope handler shall be the same as stored in buffer 1	

ld	Description	API/Framework Expectation	APDU Expectation
18	Envelope Handler integrity checks with EVENT_EVENT_DOWNLOAD_BROWSER_TERM INATION		
	1-A event download browser termination envelope is sent to SIM	1- Applet is triggered	
	2-EnvelopeHandler.getTheHandler() method is called	2-No exception is thrown.	
	3-Copy the contents of the envelope handler in buffer 1 using EnvelopeHandler.copy()	3-No exception is thrown.	
	The EnvelopeHandler.findTLV() method is called with TAG_EVENT_LIST 4-A proactive command DISPLAY TEXT is sent		
	5-Envelope call control by sim is sent to SIM	5- Applet is triggered	4-91 XX
	EnvelopeHandler.getTheHandler() method is called		
	6-It's checked that the contents of the envelope handler is the envelope call control using EnvelopeHandler.copy() and Util.arrayCompare() methods	6- No exception is thrown and the handler contains the envelope call control by SIM	
	The EnvelopeHandler.findTLV() method is called with TAG_DEVICE_IDENTITIES		
	Call Control execution is finished.		
			Proactive command Display Text is fetched
			The terminal Response of DISPLAY TEXT is sent to the SIM
	It's checked that the TAG_EVENT_LIST is the TLV selected		
	7- The contents of EnvelopeHandler are compared with buffer1 using Util.arrayCompare()	7- The contents of the envelope handler shall be the same as stored in buffer 1	

ld	Description	API/Framework Expectation	APDU Expectation
19	Envelope Handler integrity checks with UNRECOGNIZED_ENVELOPE		
	1-A unrecognized envelope is sent to SIM	1- Applet is triggered	
	2-EnvelopeHandler.getTheHandler() method is called	2- No exception is thrown.	
	3-Copy the contents of the envelope handler in buffer 1 using EnvelopeHandler.copy()	3- No exception is thrown.	
	4-A proactive command DISPLAY TEXT is sent 5-Envelope call control by sim is sent to SIM	5- Applet is triggered	4- 91 XX
		5- Applet is inggered	
	EnvelopeHandler.getTheHandler() method is called The EnvelopeHandler.getValueLength() is called		
	6-It's checked that the contents of the envelope handler is the envelope call control using EnvelopeHandler.copy() and Util.arrayCompare() methods	6- No exception is thrown and the handler contains the envelope call control by SIM	
	The EnvelopeHandler.findTLV() method is called with TAG_DEVICE_IDENTITIES		
	Call Control execution is finished.		
			Proactive command Display Text is fetched
			The terminal Response of DISPLAY TEXT is sent to the SIM
	7- The contents of EnvelopeHandler are compared with bufferl using Util.arrayCompare()	7- The contents of the envelope handler shall be the same as stored in buffer 1	

349

ld	Description	API/Framework Expectation	APDU Expectation
20	Envelope Handler integrity checks with EVENT_FORMATTED_SMS_PP_UPD		
	1-Update Record EFsms instruction single and formatted is sent to the SIM	1- Applet is triggered	
	2-EnvelopeHandler.getTheHandler() method is called	2- No exception is thrown.	
	3-Copy the contents of the envelope handler in buffer 1 using EnvelopeHandler.copy()		
	The EnvelopeHandler.findTLV() method is called with TAG_SMS_TPDU	3- No exception is thrown.	
	4-A proactive command DISPLAY TEXT is sent		4- 91 XX
	5-Envelope call control by sim is sent to SIM	5- Applet is triggered	
	EnvelopeHandler.getTheHandler() method is called		
	6-It's checked that the contents of the envelope handler is the envelope call control using EnvelopeHandler.copy() and Util.arrayCompare methods	6- No exception is thrown and the	
	The EnvelopeHandler.findTLV() method is called with TAG_SMS_TPDU	handler contains the envelope call control by SIM	
	Call Control execution is finished.		
	This should that the mid over more in		Proactive command Display Text is fetched The terminal Response of DISPLAY TEXT is sent to
	It's checked that the TAG_SMS_TPDU is the TLV selected		the SIM
	7- The contents of EnvelopeHandler are compared with buffer1 using Util.arrayCompare()	7- The contents of the envelope handler shall be the same as stored in buffer 1	

ld	Description	API/Framework Expectation	APDU Expectation
21	Envelope Handler integrity checks with EVENT_UNFORMATTED_SMS_PP_UPD		
	1- Update Record EFsms instruction single and unformatted is sent to the SIM	1- Applet is triggered	
	2-EnvelopeHandler.getTheHandler() method is called	2- No exception is thrown.	
	3-Copy the contents of the envelope handler in buffer 1 using EnvelopeHandler.copy()	3- No exception is thrown.	
	The EnvelopeHandler.findTLV method is called with TAG_SMS_TPDU		4- 91 XX
	4-A proactive command DISPLAY TEXT is sent		
	5-Envelope call control by sim is sent to SIM	5- Applet is triggered	
	EnvelopeHandler.getTheHandler() method is called		
	6-It's checked that the contents of the envelope handler is the envelope call control using EnvelopeHandler.copy() and Util.arrayCompare() methods	6- No exception is thrown and the handler contains the envelope call control by SIM	
	The EnvelopeHandler.findTLV() method is called with TAG_DEVICE_IDENTITIES		
	Call Control execution is finished.		
			Proactive command Display Text is fetched
			The terminal Response of DISPLAY TEXT is sent to the SIM
	It's checked that the TAG_DEVICE_IDENTITIES is the TLV selected		
	7- The contents of EnvelopeHandler are compared with bufferl using Util.arrayCompare()	7- The contents of the envelope handler shall be the same as stored in buffer 1.	

ld	Description	API/Framework Expectation	APDU Expectation
22	Check the TLV list conversion for EVENT_FORMATTED_SMS_PP_UPD		
	1- An EVENT_FORMATTED_SMS_PP_UPD is sent to the SIM.	1- Applet is triggered	
	2- The lindiby(tag == device identities	2- No exception is thrown.	
		3- return the absolute record.	
		4- return the record status	
	4- The getValueByte(offset == 1) is called.	5- No exception is thrown.	
	5- The findTLV(tag == address Tag) is called.		
	6- Check the content	7. No expendion is thrown	
	7- The findTLV(tag == SMS TPDU Tag) is called.	7- No exception is thrown.	
	8- Check the content		

ld	Description	API/Framework Expectation	APDU Expectation
23	Check TLV list conversion for		
	EVENT_FORMATTED_SMS_PP_UPD		
	1- The getLength() method is called	return the Simple TLV list length	

ld	Description	API/Framework Expectation	APDU Expectation
24	Check TLV list conversion for		
	EVENT_FORMATTED_SMS_PP_UPD		
	1- The getEnvelopeTag() method is called	1- return BTAG_SMS_PP_DOWNLOAD	

ld	Description	API/Framework Expectation	APDU Expectation
25	Check the TLV list conversion for EVENT_UNFORMATTED_SMS_PP_UPD		
	1- An EVENT_UNFORMATTED_SMS_PP_UPD is sent to the SIM.	1- Applet is triggered	
	2- The findTLV(tag == device identities	2- No exception is thrown.	
	Tag) is called.	3- return the absolute record.	
	<pre>3- The getValueByte(offset == 0) is called. 4- The getValueByte(offset == 1) is called.</pre>	4- return the record status	
	5- The findTLV(tag == address Tag) is called.	5- No exception is thrown.	
	6- Check the content		
	7- The findTLV(tag == SMS TPDU Tag) is called.	7- No exception is thrown.	
	8- Check the content		

ld	Description	API/Framework Expectation	APDU Expectation
26	Check TLV list conversion for		
	EVENT_UNFORMATTED_SMS_PP_UPD		
	1- The getLength() method is called	return the Simple TLV list length	

ld	Description	API/Framework Expectation	APDU Expectation
27	Check TLV list conversion for		
	EVENT_UNFORMATTED_SMS_PP_UPD		
	1- The getEnvelopeTag() method is called	1- return BTAG_SMS_PP_DOWNLOAD	

6.3.2.3.4 Test Coverage

CRR Number	Test Case Number
CRRN1	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18,
	19, 20, 21
CRRN2	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18,
	19, 20, 21
CRRN3	22, 25
CRRN4	23, 26
CRRN5	24, 27
CRRN6	22, 25

6.3.2.4 EnvelopeResponseHandler

Test Area Reference: FWK_HIN_ERHD

6.3.2.4.1 Conformance Requirement

6.3.2.4.1.1 Normal Execution

• CRRN1: At the processToolkit invocation the TLV-List is cleared.

6.3.2.4.2 Test Suite Files:

Test Script: FWK_HIN_ERHD_1.scr

Test Applet: FWK_HIN_ERHD_1.java

Load Script: FWK_HIN_ERHD_1.ldr

Cleanup Script: FWK_HIN_ERHD_1.clr

Parameter File: FWK_HIN_ERHD_1.par

6.3.2.4.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1	Applet1 is registered to EVENT_UNRECOGNIZED_ENVELOPE.		
	1-An unrecognised envelope is sent to the SIM	1- Applet 1 is triggered.	
	2- EnvelopeResponseHandler.getTheHandler()is called by the Applet1.		
	<pre>3- EnvelopeResponseHandler.getLength() method is called by Applet1</pre>	2- The return value shall be 0.	

6.3.2.4.4 Test Coverage

CRR Number	Number Test Case Number	
CRRN1	1	

6.3.3 Applet Triggering

6.3.3.1 EVENT_PROFILE_DOWNLOAD

Test Area Reference: FWK_APT_EPDW

6.3.3.1.1 Conformance Requirement

6.3.3.1.1.1 Normal Execution

- CRRN1: Upon the reception of Terminal Profile command by the SIM, the STF stores the ME Profile and then triggers the registered toolkit applets.
- CRRN2: The applet is not triggered by the EVENT_PROFILE_DOWNLOAD once it has deregistered from this event.
- CRRN3: The STF shall not reply busy to a Terminal Profile command

6.3.3.1.1.2 Parameters error

No requirements.

6.3.3.1.1.3 Context Errors

No requirements.

6.3.3.1.2 Test Suite Files

Test Script: FWK_APT_EPDW_1.scr

Test Applet: FWK_APT_EPDW_1.java

FWK_APT_EPDW_2.java

FWK_APT_EPDW_3.java

Load Script: FWK_APT_EPDW_1.ldr

Cleanup Script: FWK_APT_EPDW_1.clr

Parameter File: FWK_APT_EPDW_1.par

6.3.3.1.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1	Applets registration to EVENT_PROFILE_DOWNLOAD and triggering		
	Applet1 is registered to the EVENT_PROFILE_DOWNLOAD		
	Applet2 is registered to the EVENT_PROFILE_DOWNLOAD		
	Applet3 is not registered to the EVENT_PROFILE_DOWNLOAD and is registered to EVENT_FORMATTED_SMS_PP_ENV.		
	1-Terminal Profile command is sent to SIM	1- Applet1 is triggered	
		Applet1 finalizes 2- Applet2 is triggered	

ld	Description	API/Framework Expectation	APDU Expectation
		Applet2 finalizes 3- Applet3 is not triggered	
2	The STF shall not reply busy to a Terminal Profile command		
	1-Formatted sms pp envelope is sent to SIM	1- Applet3 is triggered by the EVENT_FORMATTED_SMS_PP_ENV	
	Applet3 builds a REFRESH proactive command in sim initialization mode 2-ProactiveHandler.send() method is called by applet3		2- A proactive command is
		Applet3 is suspended until the terminal response	sent
	3-Terminal Profile command is sent to SIM	3- Applet1 is triggered by EVENT_PROFILE_DOWNLOAD	
	Applet1 calls Toolkit Registry.clearEvent(EVENT_PROFILE_DOWNLOAD)		
	4-Applet2 calls Toolkit Registry.clearEvent(EVENT_PROFILE_DOWNLOAD) ToolkitRegistry.setEvent(EVENT_PROFILE_DOWNLOAD) method is called	Applet1 finalizes 4- Applet2 is triggered by EVENT_PROFILE_DOWNLOAD	The terminal Response of the proactive command is sent
		Applet2 finalizes	
		Applet3 finalizes	
3	Deregistered applets are not triggered Terminal Profile command is sent to SIM	Applet3 is triggered (Applet1 and Applet2 are not triggered)	

6.3.3.1.4 Test Coverage

CRR Number	Test Case Number	
CRRN1	1	
CRRN2	3	
CRRN3	2	

6.3.3.2 EVENT_MENU_SELECTION

Test Area Reference: FWK_APT_EMSE

6.3.3.2.1 Conformance Requirement

6.3.3.2.1.1 Normal Execution

• CRRN1: The applet is triggered by the EVENT_MENU_SELECTION when an Envelope Menu Selection is received with the item identifier of a menu entry of this applet if no proactive session is ongoing.

6.3.3.2.1.2 Parameters error

No requirements.

6.3.3.2.1.3 Context Errors

No requirements.

6.3.3.2.2 Test Suite Files

Test Script: FWK_APT_EMSE_1.scr

Test Applet: FWK_APT_EMSE_1.java

FWK_APT_EMSE_2.java

Load Script: FWK_APT_EMSE_1.ldr

Cleanup Script: FWK_APT_EMSE_1.clr

Parameter File: FWK_APT_EMSE_1.par

6.3.3.2.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1d 1	Description Applet registration to EVENT_MENU_SELECTION and triggering ToolkitRegistry.initMenuEntry() method is called in the constructor of applet1 and Applet2. For applet1: MenuEntry="Applet1" Offset=0 Length=menuEntry.length HelpSupported=false IconQualifier=0 IconIdentifier=0 For applet2: MenuEntry="Applet2" Offset=0 Length=menuEntry.length HelpSupported=false IconQualifier=0 Length=menuEntry.length HelpSupported=false IconQualifier=0 Length=menuEntry.length HelpSupported=false IconQualifier=0 LoonIdentifier=0 event= EVENT_MENU_SELECTION 1-ToolkitRegistry.isEventSet() is called	1- The method must return true.	APDU Expectation
	in constructor. Perform SIM initialization the facility SET UP MENU and without the facilities SET EVENT LIST and POLL INTEVAL features 2-Item Identifier = 1 Event Menu Selection envelope is sent to the SIM with the item identifier of a menu entry of applet 3-Item Identifier = 2 Event Menu Selection envelope is sent to the SIM with the item identifier of a menu entry of applet	2- Applet1 is triggered and applet2 is not triggered Applet1 finalizes 3- Applet2 is triggered and applet1 is not triggered	

6.3.3.2.4 Test Coverage

CRR Number	Test Case Number
CRRN1	1

6.3.3.3 EVENT_MENU_SELECTION_HELP_REQUEST

Test Area Reference: FWK_APT_EMSH

6.3.3.3.1 Conformance Requirement

6.3.3.3.1.1 Normal Execution

- CRRN1: If an ENVELOPE (MENU_SELECTION_HELP_SUPPORTED) command is received for one entry supporting help, then STF shall trigger the corresponding applet.
- CCRN2: A toolkit applet shall be triggered by the EVENT_MENU_SELECTION_HELP_REQUEST event only if the Menu Id corresponding to the Envelope Menu Selection Help Request received by the SIM Toolkit framework was registered with the helpSupported value set to true.
- CCRN3: If at least one menuId of a Toolkit Applet registers to EVENT_MENU_SELECTION_HELP_REQUEST, the SET UP MENU proactive command sent by the SIM Toolkit Framework shall indicate to the ME that help information is available unless all the menus entries that support help are disabled.

6.3.3.3.1.2 Parameters error

No requirements.

6.3.3.3.1.3 Context Errors

No requirements.

6.3.3.3.2 Test Suite Files

Test Script: FWK_APT_EMSH_1.scr

Test Applet: FWK_APT_EMSH_1.java

FWK_APT_EMSH_2.java

FWK_APT_EMSH_3.java

Load Script: FWK_APT_EMSH_1.ldr

Cleanup Script: FWK_APT_EMSH_1.clr

Parameter File: FWK_APT_EMSH_1.par

6.3.3.3.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1	Applet registration to EVENT_MENU_SELECTION_HELP_REQUEST and triggering		
	Applet1 and Applet2 are installed		
	ToolkitRegistry.InitMenuEntry() method is called in the constructor of Applet1 and Applet2.		
	For Applet1 (item id 1): MenuEntry="Applet1A" Offset=0 Length=menuEntry.length HelpSupported=true IconQualifier=0 IconIdentifier=0		
	For Applet1 (item id 2): MenuEntry="Applet1B" Offset=0		
	Length=menuEntry.length HelpSupported=false IconQualifier=0 IconIdentifier=0		
	<pre>event= EVENT_MENU_SELECTION_HELP_REQUEST 1- ToolkitRegistry.isEventSet() is called in constructor.</pre>		
		1- The command shall return true.	
	For Applet2 (item id 3): MenuEntry="Applet2A" Offset=0 Length=menuEntry.length HelpSupported=true IconQualifier=0 IconIdentifier=0		
	For Applet2 (item id 4): MenuEntry="Applet2B" Offset=0 Length=menuEntry.length HelpSupported=false IconQualifier=0 IconIdentifier=0		
	<pre>event= EVENT_MENU_SELECTION_HELP_REQUEST 2- ToolkitRegistry.isEventSet() is called in constructor.</pre>	2- The command shall return true.	
	Perform SIM initialization with the facility SET UP MENU and without the facilities SET EVENT LIST and POLL INTERVAL		
	3-Item identifier = 1 Menu Selection Help Request envelope is sent to the SIM with item identifier 1 belonging to applet1	3- Applet1 is triggered and Applet2 is not triggered	
	4-Item identifier = 2 Menu Selection Help Request envelope is sent to the SIM with item identifier 2 belonging to applet1	4 Applet1 and Applet2 are not triggered	

ld	Description	API/Framework Expectation	APDU Expectation
	5-Item identifier = 3 Menu Selection Help Request envelope is sent to the SIM with item identifier 3 belonging to applet2	5- Applet2 is triggered and Applet1 is not triggered	
	6-Item identifier = 4 Menu Selection Help Request envelope is sent to the SIM with item identifier 4 belonging to applet2	6- Applet2 and Applet1 are not triggered	

ld	Description	API/Framework Expectation	APDU Expectation
2	Applet deregistration to		
	EVENT_MENU_SELECTION_HELP_REQUEST		
	Applet1 and Applet2 are deleted		
	Applet3 is installed		
	ToolkitRegistry.InitMenuEntry() method is called in the constructor of Applet3.		
	For Applet3 (item id 5): MenuEntry="Applet3A" Offset=0 Length=menuEntry.length HelpSupported=true IconQualifier=0		
	IconIdentifier=0		
	For Applet3 (item id 6): MenuEntry="Applet3B" Offset=0 Length=menuEntry.length HelpSupported=true IconQualifier=0 IconIdentifier=0		
	For Applet3 (item id 7): MenuEntry="Applet3C" Offset=0 Length=menuEntry.length HelpSupported=false IconQualifier=0 IconIdentifier=0		
	1. Perform SIM initialization with the facility SET UP MENU and without the facilities SET EVENT LIST and POLL INTERVAL		1. The SIM shall issue a SET UP MENU proactive command with Menu Entry ID entry '05', '06' and '07',
	2. Menu Selection Help Request envelope is sent to the SIM with item identifier 5 belonging to applet3		and Help supported set to true.
	3. ToolkitRegistry.disableMenuEntry() method for item id 5 is called by the Menu Selection Help Request Envelope.	Applet3 is triggered by EVENT_MENU_SELECTION_HEL	3. The SIM shall issue a
	4. Menu Selection Help Request envelope is sent to the SIM with item identifier 6 belonging to applet3	P_REQUEST	SET UP MENU proactive command with Menu Entry ID entry '06' and '07', and Help supported set to true.
	5. ToolkitRegistry.disableMenuEntry() method for item id 6 is called by the Menu Selection Help Request Envelope.	4. Applet3 is triggered by EVENT_MENU_SELECTION_HEL P_REQUEST	
			5. The SIM shall issue a SET UP MENU proactive command with Menu Entry ID entry '07', and Help supported set to false.

6.3.3.3.4 Test Coverage

CRR Number	Test Case Number

CRR Number	Test Case Number
CRRN1	1
CRRN2	1
CRRN3	2

6.3.3.4 EVENT_FORMATTED_SMS_PP_ENV

Test Area Reference: FWK_APT_EFSE

6.3.3.4.1 Conformance Requirement

6.3.3.4.1.1 Normal Execution

- CRRN1: The applet is triggered by the EVENT_FORMATTED_SMS_PP_ENV once:
 - it has been registered to this event;
 - a Short Message Point to Point (Single or Concatenated) is received by Envelope APDU(s) and is formatted according to 3GPP TS 23.048 [8];
 - the toolkit applet to be triggered is registered with the corresponding TAR in the SMS TPDU;
 - the security is verified.
- CRRN2: The applet is not triggered by the EVENT_FORMATTED_SMS_PP_ENV once it has deregistered
 from this event.

6.3.3.4.1.2 Parameters error

No requirements.

6.3.3.4.1.3 Context Errors

No requirements.

6.3.3.4.2 Test Suite Files

Test Script: FWK_APT_EFSE_1.scr

Test Applet: FWK_APT_EFSE_1.java

Load Script: FWK_APT_EFSE_1.ldr

Cleanup Script: FWK_APT_EFSE_1.clr

Parameter File: FWK_APT_EFSE_1.par

6.3.3.4.3 Test Procedure

le	l Description	API/Framework Expectation	APDU Expectation
1	Applet registration to EVENT FORMATTED_SMS_PP_ENV and triggering		
	Applet is registered to EVENT_FORMATTED_SMS_PP_ENV and EVENT_UNRECOGNIZED_ENVELOPE		
	1- A Single Short Message SMS-PP Formatted Data Download is sent to the SIM. 2- A Concatenated Short Message SMS-PP		

ld	Description	API/Framework Expectation	APDU Expectation
	Formatted Data Download is sent to the SIM (composed of 2 Short Messages. The UDL for the first Short Message is 70 and for the second 70)	1- Applet is triggered2- Applet is triggered	
2	Applet deregistration		
	ToolkitRegistry.clearEvent() method is called for EVENT_FORMATTED_SMS_PP_ENV 1- A Single Short Message SMS-PP Data Download is sent to the SIM 2- A Concatenated Short Messages SMS-PP Data Download is sent to the SIM (composed of 2 Short Messages. The UDL for the first Short Message is 70 and for the second 70).		
		2- Applet is not triggered	
	An unrecognized envelope is sent to the sim		
	ToolkitRegistry.setEvent() method is called for EVENT_FORMATTED_SMS_PP_ENV		
	3- A Single Short Messages SMS-PP Data Download is sent to the SIM.		
	4- A Concatenated Short Messages SMS-PP Data Download is sent to the SIM (composed of 2 Short Messages. The UDL for the first Short Message is 70 and for the second	3- Applet is triggered	
	70).	4- Applet is triggered	

6.3.3.4.4 Test Coverage

CRR Number	Test Case Number
CRRN1 (See note)	1, 2
CRRN2	2

NOTE: The security checks are not relevant to the test designed in this test area; they will be checked in the "Framework Security Management" section.

6.3.3.5 EVENT_UNFORMATTED_SMS_PP_ENV

Test Area Reference: FWK_APT_EUSE

6.3.3.5.1 Conformance Requirement

6.3.3.5.1.1 Normal Execution

- CRRN1: The applets registers are triggered by the EVENT_UNFORMATTED_SMS_PP_ENV once a Short Message Point to Point (Single or Concatenated) is received by Envelope APDU(s) and is unformatted.
- CRRN2: The applet is not triggered by the EVENT_UNFORMATTED_SMS_PP_ENV once it has deregistered from this event.

6.3.3.5.1.2 Parameters error

No requirements.

6.3.3.5.1.3 Context Errors

No requirements.

6.3.3.5.2 Test Suite Files

Test Script: FWK_APT_EUSE_1.scr

Test Applet: FWK_APT_EUSE_1.java

Load Script: FWK_APT_EUSE_1.ldr

Cleanup Script: FWK_APT_EUSE_1.clr

Parameter File: FWK_APT_EUSE_1.par

6.3.3.5.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1	Applet registration to EVENT_UNFORMATTED_SMS_PP_ENV and triggering		-
	Applet is registered to the EVENT_UNFORMATTED_SMS_PP_ENV and EVENT_FORMATTED_SMS_PP_ENV.		
	1-Toolkit Registry.isEventSet() method is called for EVENT_UNFORMATTED_SMS_PP_ENV	1- The method returns true	
	2- A Single and Unformatted SMS-PP Data Download Envelope is sent to the SIM.	2- Applet is triggered	
	3- A Concatenated and Unformatted SMS-PP Data Download Envelope is sent to the SIM (composed of 2 Short Messages. The UDL for the first Short Message is 70 and for the second 70)	3- Applet is triggered	

ld	Description	API/Framework Expectation	APDU Expectation
2	Applet deregistration		
	Toolkit Registry.clearEvent()method is called for EVENT_UNFORMATTED_SMS_PP_ENV		
	1- A Single and Unformatted SMS-PP Data Download Envelope is sent to the SIM.	1- Applet isn't triggered	
	2- A Concatenated and Unformatted SMS-PP Data Download Envelope is sent to the SIM (composed of 2 Short Messages. The UDL for the first Short Message is 70 and for the second 70)	2- Applet isn't triggered	
	Applet is triggered by a EVENT_FORMATTED_SMS_PP_ENV		
	Toolkit Registry.setEvent() method is called for EVENT_UNFORMATTED_SMS_PP_ENV		
	3- A Single and Unformatted SMS-PP Data Download Envelope is sent to the SIM.	3- Applet is triggered	
	4- A Concatenated and Unformatted SMS-PP Data Download Envelope is sent to the SIM (composed of 2 Short Messages. The UDL for the first Short Message is 70 and for the second 70)	4- Applet is triggered	

6.3.3.5.4 Test Coverage

CRR Number	Test Case Number
CRRN1	1, 2
CRRN2	2

6.3.3.6 EVENT_CALL_CONTROL_BY_SIM

Test Area Reference: FWK_APT_ECCN

6.3.3.6.1 Conformance Requirement

6.3.3.6.1.1 Normal Execution

• CRRN1: The applet is triggered by the EVENT_CALL_CONTROL_BY_SIM once it has registered to this event and an Envelope Call Control is received.

• CRRN2: The applet is not triggered by the EVENT_CALL_CONTROL_BY_SIM once it has deregistered from this event.

6.3.3.6.1.2 Parameters error

No requirements.

6.3.3.6.1.3 Context Errors

No requirements.

6.3.3.6.2 Test Suite Files

Test Script: FWK_APT_ECCN_1.scr

Test Applet: FWK_APT_ECCN_1.java

Load Script: FWK_APT_ECCN_1.ldr

Cleanup Script: FWK_APT_ECCN_1.clr

Parameter File: FWK_APT_ECCN_1.par

6.3.3.6.3 Test Procedure

	Description	API/Framework Expectation	APDU Expectation
1	Applets registration to EVENT_CALL_CONTROL_BY_SIM and triggering		
	Applet1 is registered to EVENT_CALL_CONTROL_BY_SIM.		
	Applet2 is registered to EVENT_FORMATTED_SMS_PP_ENV		
	1-An Envelope Call control by SIM is sent to SIM	1- Applet1 is triggered	
2	Applet deregistration and registration of the third applet to EVENT_CALL-CONTROL_BY_SIM.		
	1-An Envelope Formatted SMS PP envelope is sent to SIM	1-Applet2 is triggered by EVENT_FORMATTED_SMS_PP_ENV.	
	Applet2 contructs a DISPLAY TEXT proactive command.		
	2-ProactiveHandler.send() method is called		2- A proactive command DISPLAY TEXT is sent and applet is suspended until the terminal response
	3-An Envelope Call control by SIM envelope is sent to SIM	3- Applet1 is triggered	
	ToolkitRegistry.clearEvent() is called for EVENT_CALL_CONTROL_BY_SIM.		
		Applet1 finalizes.	TERMINAL RESPONSE of DISPLAY TEXT is sent to the SIM
	ToolkitRegistry.setEvent() method is called for EVENT_CALL_CONTROL_BY_SIM.		
		Applet2 finalizes	
3	Applet triggering	, .ps.c	
	An Envelope Call control by SIM envelope is sent ot SIM	Applet2 is triggered. (Applet1 is not triggered)	

6.3.3.6.4 Test Coverage

CRR Number	Test Case Number
CRRN1	1, 2, 3
CRRN2	3

6.3.3.7 EVENT_MO_SHORT_MESSAGE_CONTROL_BY_SIM

Test Area Reference: FWK_APT_EMCN

6.3.3.7.1 Conformance Requirement

6.3.3.7.1.1 Normal Execution

- CRRN1: The applet is triggered by the EVENT_MO_SHORT_MESSAGE_CONTROL_BY_SIM once it has registered to this event and an Envelope MO Short Message Control.
- CRRN2: The applet is not triggered by the EVENT_MO_SHORT_MESSAGE_CONTROL_BY_SIM once it has deregistered from this event.

6.3.3.7.1.2 Parameters error

No requirements.

6.3.3.7.1.3 Context Errors

No requirements.

6.3.3.7.2 Test Suite Files

Test Script: FWK_APT_EMCN_1.scr

Test Applet: FWK_APT_EMCN_1.java

FWK_APT_EMCN_2.java

Load Script: FWK_APT_EMCN_1.ldr

Cleanup Script: FWK_APT_EMCN_1.clr

Parameter File: FWK_APT_EMCN_1.par

6.3.3.7.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1	Applet registration to EVENT_MO_SHORT_MESSAGE_CONTROL_B Y_SIM and triggering		
	Applet1 is reggistered to EVENT_MO_SHORT_MESSAGE_CONTROL_BY_SIM.		
	Applet2 is registered to EVENT_FORMATTED_SMS_PP_ENV.		
	1-An Envelope MO short message envelope is sent to SIM	1- Applet1 is triggered.	

ld	Description	API/Framework Expectation	APDU Expectation
2	Applet deregistration and registration of the third applet to EVENT_MO_SHORT_MESSAGE_CONTROL_B Y_SIM. The STF shall not reply busy to a call control envelope		
	1-An Envelope formatted SMS PP envelope is sent to SIM.	1- Applet2 is triggered.	
	Applet2 builds a DISPLAY TEXT proactive command. 2-ProactiveHandler.send() method is		2- A Proactive command DISPLAY TEXT is sent and
	called.		applet is suspended until the terminal response
	3-An Envelope MO Short message envelope is sent to SIM	3- Applet1 is triggered.	
	ToolkitRegistry.clearEvent() for EVENT_MO_SHORT_MESSAGE_CONTROL_BY_SIM.		TERMINAL RESPONSE of
		Applet1 finalizes.	DISPLAY TEXT is sent to the SIM
	ToolkitRegistry.setEvent() method is called for EVENT_MO_SHORT_MESSAGE_CONTROL_BY_SIM.		
		Applet2 finalizes.	
3	Applet3 triggering		
	An Envelope MO SMS control by SIM envelope is sent ot SIM	Applet2 is triggered. (Applet1 is not triggered)	

6.3.3.7.4 Test Coverage

CRR Number	Test Case Number
CRRN1	1, 2, 3
CRRN2	3

6.3.3.8 EVENT_TIMER_EXPIRATION

Test Area Reference: FWK_APT_ETEX

6.3.3.8.1 Conformance Requirement

6.3.3.8.1.1 Normal Execution

- CRRN1: The applet is triggered by the EVENT_TIMER_EXPIRATION once it has been registered to this event and an Envelope Timer Expiration with a Timer Identifier of the applet is received if no proactive session is ongoing.
- CRRN2: The applet is not triggered by the EVENT_TIMER_EXPIRATION once it has been deregistered from this event.

6.3.3.8.1.2 Parameters error

No requirements.

6.3.3.8.1.3 Context Errors

No requirements.

6.3.3.8.2 Test Suite Files

Test Script: FWK_APT_ETEX_1.scr

Test Applet: FWK_APT_ETEX_1.java

Load Script: FWK_APT_ETEX_1.ldr

Cleanup Script: FWK_APT_ETEX_1.clr

Parameter File: FWK_APT_ETEX_1.par

6.3.3.8.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1	Applet registration to EVENT_TIMER_EXPIRATION and triggering Applet is registered to the EVENT_TIMER_EXPIRATION using the allocateTimer() method and to EVENT_FORMATTED_SMS_PP_ENV.		
	<pre>event= EVENT_TIMER_EXPIRATION 1-Toolkit Registry.isEventSet() method is called.</pre>	1- The method returns true	
	2-An Envelope TIMER_EXPIRATION is sent to the SIM.	2- Applet is triggered.	
2	Applet deregistration		
	Timer id=1 Toolkit Registry.ReleaseTimer() method is called 1-An Envelope timer expiration is sent to the SIM.	1- Applet isn't triggered	
	An Envelope formated sms pp envelope is sent to the sim		
	Toolkit Registry.AllocateTimer() method is called		
	2-An Envelope TIMER_EXPIRATION is sent to the SIM.	2- Applet is triggered	

6.3.3.8.4 Test Coverage

CRR Number	Test Case Number	
CRRN1	1, 2	
CRRN2	2	

6.3.3.9 EVENT_UNFORMATTED_SMS_CB

Test Area Reference: FWK_APT_EUCB

6.3.3.9.1 Conformance Requirement

6.3.3.9.1.1 Normal Execution

- CRRN1: The applet is triggered by the EVENT_UNFORMATTED_SMS_CB once it has registered to this event and an Envelope Cell Broadcast DownLoad is received.
- CRRN2: The applet is not triggered by the EVENT_UNFORMATTED_SMS_CB once it has deregistered from this event.

6.3.3.9.1.2 Parameters error

No requirements.

6.3.3.9.1.3 Context Errors

No requirements.

6.3.3.9.2 Test Suite Files

Test Script: FWK_APT_EUCB_1.scr

Test Applet: FWK_APT_EUCB_1.java

Load Script: FWK_APT_EUCB_1.ldr

Cleanup Script: FWK_APT_EUCB_1.clr

Parameter File: FWK_APT_EUCB_1.par

6.3.3.9.3 Test Procedure

ld	Description	API Expectation	APDU Expectation
1	Applet registration to EVENT_UNFORMATTED_SMS_CB and triggering		-
	Applet is registered to the EVENT_UNFORMATTED_SMS_CB and EVENT_FORMATTED_SMS_PP_ENV.		
	<pre>event= EVENT_UNFORMATTED_SMS_CB 1-Toolkit Registry.isEventSet() method is called.</pre>	1- Method returns true.	
	2-An Envelope UNFORMATTED_SMS_CB is sent to the SIM.	2- Applet is triggered	

ld	Description	API Expectation	APDU Expectation
2	Applet deregistration		
	Toolkit Registry.ClearEvent()method is called for EVENT_UNFORMATTED_SMS_CB		
	1-An Envelope UNFORMATTED_SMS_CB is sent to the SIM.	1- Applet isn't triggered	
	An Envelope formatted sms pp envelope is sent to the sim		
	event= EVENT_UNFORMATTED_SMS_CB		
	Toolkit Registry.setEvent() method is called for EVENT_UNFORMATTED_SMS_CB		
	2-An Envelope UNFORMATTED_SMS_CB is sent to the SIM.	2- Applet is triggered	

6.3.3.9.4 Test Coverage

CRR Number	Test Case Number
CRRN1	1, 2
CRRN2	2

6.3.3.10 EVENT_EVENT_DOWNLOAD_MT_CALL

Test Area Reference: FWK_APT_EDMC

6.3.3.10.1 Conformance Requirement

6.3.3.10.1.1 Normal Execution

- CRRN1: The applet is triggered by the EVENT_EVENT_DOWNLOAD_MT_CALL once it has registered to this event and an Envelope Event DownLoad MT Call is received.
- CRRN2: The applet is not triggered by the EVENT_EVENT_DOWNLOAD_MT_CALL once it has deregistered from this event.

6.3.3.10.1.2 Parameters error

No requirements.

6.3.3.10.1.3 Context Errors

No requirements.

6.3.3.10.2 Test Suite Files

Test Script: FWK_APT_EMSE_1.scr

Test Applet: FWK_APT_EMSE_1.java

Load Script: FWK_APT_EMSE_1.ldr

Cleanup Script: FWK_APT_EMSE_1.clr

Parameter File: FWK_APT_EMSE_1.par

6.3.3.10.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1	Applet registration to EVENT_EVENT_DOWNLOAD_MT_CALL and triggering		
	Applet is registered to the EVENT_EVENT_DOWNLOAD_MT_CALL and to EVENT_FORMATTED_SMS_PP_ENV.		
	event= EVENT_EVENT_DOWNLOAD_MT_CALL 1-Toolkit Registry.isEventSet() method is called.	1- The method returns true	
	2-An Envelope EVENT_DOWNLOAD_MT_CALL is sent to the SIM.	2- Applet is triggered	
2	Applet deregistration		
	event= EVENT_EVENT_DOWNLOAD_MT_CALL Toolkit Registry.clearEvent()method is called Perform SIM initialization with all the facilities supported 1-An Envelope EVENT_DOWNLOAD_MT_CALL is sent to the SIM.	1- Applet isn't triggered	
	An Envelope formatted sms pp envelope is sent to the sim event= EVENT_EVENT_DOWNLOAD_MT_CALL Toolkit Registry.setEvent() method is called		
	Perform SIM initialization with all the facilities supported		
	2-An Envelope EVENT_DOWNLOAD_MT_CALL is sent to the SIM.	2- Applet is triggered	

6.3.3.10.4 Test Coverage

CRR Number	Test Case Number
CRRN1	1, 2
CRRN2	2

6.3.3.11 EVENT_EVENT_DOWNLOAD_CALL_CONNECTED

Test Area Reference: FWK_APT_EDCC

6.3.3.11.1 Conformance Requirement

6.3.3.11.1.1 Normal Execution

• CRRN1: The applet is triggered by the EVENT_EVENT_DOWNLOAD_CALL_CONNECTED once it has registered to this event and an Envelope Event DownLoad Call Connected is received.

• CRRN2: The applet is not triggered by the EVENT_EVENT_DOWNLOAD_CALL_CONNECTED once it has deregistered from this event.

6.3.3.11.1.2 Parameters error

No requirements.

6.3.3.11.1.3 Context Errors

No requirements.

6.3.3.11.2 Test Suite Files

Test Script: FWK_APT_EDCC_1.scr

Test Applet: FWK_APT_EDCC_1.java

Load Script: FWK_APT_EDCC_1.ldr

Clean-up Script: FWK_APT_EDCC_1.clr

Parameter File: FWK_APT_EDCC _1.par

6.3.3.11.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1	Applet registration to EVENT_EVENT_DOWNLOAD_CALL_CONNECT ED and triggering		
	Applet is registered to the EVENT_EVENT_DOWNLOAD_CALL_CONNECTED and to EVENT_FORMATTED_SMS_PP_ENV.		
	<pre>event= EVENT_EVENT_DOWNLOAD_CALL_CONNECTED 1-Toolkit Registry.isEventSet() method is called.</pre>	1- Method returns true	
	2-An Envelope EVENT_DOWNLOAD_CALL_CONNECTED is sent to the SIM.	2- Applet is triggered.	
2	Applet deregistration		
	event=EVENT_EVENT_DOWNLOAD_CALL_CONNECTED Toolkit Registry.clearEvent()method is called Perform SIM initialization with all the facilities supported 1-A call connected event dowload is sent to the SIM.	1- Applet isn't triggered	
	An Envelope formatted sms pp envelope is sent to the sim Event= EVENT_EVENT_DOWNLOAD_CALL_CONNECTED Toolkit Registry.setEvent() method is called		
	Perform SIM initialization with all the facilities supported 2-An Envelope EVENT_DOWNLOAD_CALL_CONNECTED is sent to the SIM.	2- Applet is triggered	

6.3.3.11.4 Test Coverage

CRR Number	Test Case Number
CRRN1	1, 2
CRRN2	2

6.3.3.12 EVENT_EVENT_DOWNLOAD_CALL_DISCONNECTED

Test Area Reference: FWK_APT_EDCD

6.3.3.12.1 Conformance Requirement

6.3.3.12.1.1 Normal Execution

- CRRN1: The applet is triggered by the EVENT_EVENT_DOWNLOAD_CALL_DISCONNECTED once it has registered to this event and an Envelope Event DownLoad Call Disconnected is received.
- CRRN2: The applet is not triggered by the EVENT_EVENT_DOWNLOAD_CALL_DISCONNECTED once it has deregistered from this event.

6.3.3.12.1.2 Parameters error

No requirements.

6.3.3.12.1.3 Context Errors

No requirements.

6.3.3.12.2 Test Suite Files

Test Script: FWK_APT_EDCD_1.scr

Test Applet: FWK_APT_EDCD_1.java

Load Script: FWK_APT_EDCD_1.ldr

Cleanup Script: FWK_APT_EDCD_1.clr

Parameter File: FWK_APT_EDCD_1.par

6.3.3.12.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1	Applet registration to EVENT_EVENT_DOWNLOAD_CALL_DISCONN ECTED and triggering		
	Applet is registered to the EVENT_EVENT_DOWNLOAD_CALL_DISCONNECTED and to EVENT_FORMATTED_SMS_PP_ENV.		
	Event=EVENT_EVENT_DOWNLOAD_CALL_DISCONNECT ED 1-Toolkit Registry.isEventSet() method is called.	1- Method returns true	
	2-An Envelope EVENT_EVENT_DOWNLOAD_CALL_DISCONNECTED is sent to the SIM.	2- Applet is triggered.	

ld	Description	API/Framework Expectation	APDU Expectation
2	Applet deregistration		
	Event= EVENT_EVENT_DOWNLOAD_CALL_DISCONNECTED Toolkit Registry.clearEvent()method is called Perform SIM initialization with all the facilities supported 1-An Envelope EVENT_DOWNLOAD_CALL_DISCONNECTED is sent to the SIM. a formatted sms pp envelope is sent to the	1- Applet isn't triggered	
	sim. Event= EVENT_EVENT_DOWNLOAD_CALL_DISCONNECTED Toolkit Registry.setEvent() method is called Perform SIM initialization with all the facilities supported 2-An Envelope EVENT_EVENT_DOWNLOAD_CALL_DISCONNECTED is sent to the SIM.	2- Applet is triggered	

6.3.3.12.4 Test Coverage

CRR Number	Test Case Number
CRRN1	1, 2
CRRN2	2

6.3.3.13 EVENT_EVENT_DOWNLOAD_LOCATION_STATUS

Test Area Reference: FWK_APT_EDLS

6.3.3.13.1 Conformance Requirement

6.3.3.13.1.1 Normal Execution

- CRRN1: The applet is triggered by the EVENT_EVENT_DOWNLOAD_LOCATION_STATUS once it has registered to this event and an Envelope Event DownLoad Location Status is received.
- CRRN2: The applet is not triggered by the EVENT_EVENT_DOWNLOAD_LOCATION_STATUS once it has deregistered from this event.

6.3.3.13.1.2 Parameters error

No requirements.

6.3.3.13.1.3 Context Errors

No requirements.

6.3.3.13.2 Test Suite Files

Test Script: FWK_APT_EDLS_1.scr

Test Applet: FWK_APT_EDLS_1.java

Load Script: FWK_APT_EDLS_1.ldr

Cleanup Script: FWK_APT_EDLS_1.clr

Parameter File: FWK_APT_EDLS_1.par

6.3.3.13.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1	Applet registration to EVENT_EVENT_DOWNLOAD_LOACTION_STA TUS and triggering		
	Applet is registered to the EVENT_EVENT_DOWNLOAD_LOCATION_STATUS and to EVENT_FORMATTED_SMS_PP_ENV.		
	<pre>Event=EVENT_EVENT_DOWNLOAD_LOCATION_STATUS 1-Toolkit Registry.isEventSet() method is called.</pre>	1- Method returns true	
	2-An Envelope EVENT_EVENT_DOWNLOAD_LOCATION_STATUS is sent to the SIM.	2- Applet is triggered.	
2	Applet deregistration		
_	Applet deregistration		
	Event=EVENT_EVENT_DOWNLOAD_LOCATION_STATUS Toolkit Registry.clearEvent()method is called Perform SIM initialization with all the facilities supported		
	1-An Envelope EVENT_DOWNLOAD_LOCATION_STATUS is sent to the SIM.	1- Applet isn't triggered	
	a formatted sms pp envelope is sent to the sim		
	Event= EVENT_EVENT_DOWNLOAD_LOCATION_STATUS Toolkit Registry.setEvent() method is called Perform SIM initialization with all the facilities supported		
	2-An Envelope EVENT_DOWNLOAD_LOCATION_STATUS is sent to the SIM.	2- Applet is triggered	

6.3.3.13.4 Test Coverage

CRR Number	Test Case Number
CRRN1	1, 2
CRRN2	2

6.3.3.14 EVENT_EVENT_DOWNLOAD_USER_ACTIVITY

Test Area Reference: FWK_APT_EDUA

6.3.3.14.1 Conformance Requirement

6.3.3.14.1.1 Normal Execution

- CRRN1: The applet is triggered by the EVENT_EVENT_DOWNLOAD_USER_ACTIVITY once it has registered to this event and an Envelope Event DownLoad User Activity is received.
- CRRN2: The applet is not triggered by the EVENT_EVENT_DOWNLOAD_USER_ACTIVITY once it has deregistered from this event.

6.3.3.14.1.2 Parameters error

No requirements.

6.3.3.14.1.3 Context Errors

No requirements.

6.3.3.14.2 Test Suite Files

Test Script: FWK_APT_EDUA_1.scr

Test Applet: FWK_APT_EDUA_1.java

Load Script: FWK_APT_EDUA_1.ldr

Cleanup Script: FWK_APT_EDUA_1.clr

Parameter File: FWK_APT_EDUA_1.par

6.3.3.14.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1	Applet registration to EVENT_EVENT_DOWNLOAD_USER_ACTIVITY and triggering		
	Applet is registered to the EVENT_EVENT_DOWNLOAD_USER_ACTIVITY and to EVENT_FORMATTED_SMS_PP_ENV.		
	Event= EVENT_EVENT_DOWNLOAD_USER_ACTIVITY		
	1-Toolkit Registry.isEventSet() method is called.	1- Method returns true	
	2-An Envelope EVENT_DOWNLOAD_USER_ACTIVITY is sent to the SIM.	2- Applet is triggered	

ld	Description	API/Framework Expectation	APDU Expectation
2	Applet deregistration		
	Event= EVENT_EVENT_DOWNLOAD_USER_ACTIVITY Toolkit Registry.clearEvent()method is called		
	Perform SIM initialization with all the facilities supported		
	1-An Envelope EVENT_DOWNLOAD_USER_ACTIVITY is sent to the SIM.	1- Applet isn't triggered	
	a formatted sms pp envelope is sent to the $\operatorname{\text{\rm sim}}$		
	Event= EVENT_EVENT_DOWNLOAD_USER_ACTIVITY Toolkit Registry.setEvent() method is called Perform SIM initialization with all the facilities supported		
	2-An Envelope EVENT_DOWNLOAD_USER_ACTIVITY is sent to the SIM.	2- Applet is triggered	

6.3.3.14.4 Test Coverage

CRR Number	Test Case Number
CRRN1	1, 2
CRRN2	2

6.3.3.15 EVENT_EVENT_DOWNLOAD_IDLE_SCREEN_AVAILABLE

Test Area Reference: FWK_APT_EDIS

6.3.3.15.1 Conformance Requirement

6.3.3.15.1.1 Normal Execution

- CRRN1: The applet is triggered by the EVENT_EVENT_DOWNLOAD_IDLE_SCREEN_AVAILABLE once it has registered to this event and an Envelope Event DownLoad Idle Screen Available is received.
- CRRN2: The applet is not triggered by the EVENT_EVENT_DOWNLOAD_IDLE_SCREEN_AVAILABLE once it has deregistered from this event.

6.3.3.15.1.2 Parameters error

No requirements.

6.3.3.15.1.3 Context Errors

No requirements.

6.3.3.15.2 Test Suite Files

Test Script: FWK_APT_EDIS_1.scr

Test Applet: FWK_APT_EDIS_1.java

Load Script: FWK_APT_EDIS_1.ldr

Cleanup Script: FWK_APT_EDIS_1.clr
Parameter File: FWK_APT_EDIS_1.par

6.3.3.15.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1	Applet registration to EVENT_EVENT_DOWNLOAD_IDLE_SCREEN_ AVAILABLE and triggering		
	Applet is registered to the EVENT_EVENT_DOWNLOAD_IDLE_SCREEN_AVAILABLE and to EVENT_FORMATTED_SMS_PP_ENV Event= EVENT_EVENT_DOWNLOAD_IDLE_SCREEN_AVAILABLE		
	1-Toolkit Registry.isEventSet() method is called.	1- Method returns true	
	2-An Envelope EVENT_DOWNLOAD_IDLE_SCREEN_AVAILABLE is sent to the SIM.	2- Applet is triggered	
2	Applet deregistration		
	Event=EVENT_EVENT_DOWNLOAD_IDLE_SCREEN_AVA ILABLE		
	Toolkit Registry.clearEvent()method is called Perform SIM initialization with all the facilities supported		
	1-An Envelope EVENT_DOWNLOAD_IDLE_SCREEN_AVAILABLE is sent to the SIM.	1- Applet isn't triggered	
	a formatted sms pp envelope is sent to the $\operatorname{\text{\rm sim}}$		
	Event= EVENT_EVENT_DOWNLOAD_IDLE_SCREEN_AVAILABLE		
	Toolkit Registry.setEvent() method is called		
	Perform SIM initialization with all the facilities supported		
	2-An Envelope EVENT_DOWNLOAD_IDLE_SCREEN_AVAILABLE is sent to the SIM.	2- Applet is triggered	

6.3.3.15.4 Test Coverage

CRR Number	Test Case Number
CRRN1	1, 2
CRRN2	2

6.3.3.16 EVENT_EVENT_DOWNLOAD_CARD_READER_STATUS

Test Area Reference: FWK_APT_EDCR

6.3.3.16.1 Conformance Requirement

6.3.3.16.1.1 Normal Execution

- CRRN1: The applet is triggered by the EVENT_EVENT_DOWNLOAD_CARD_READER_STATUS once it has registered to this event and Envelope Event DownLoad Card Reader Status is received.
- CRRN2: The applet is not triggered by the EVENT_EVENT_DOWNLOAD_CARD_READER_STATUS once it has deregistered from this event.

6.3.3.16.1.2 Parameters error

No requirements.

6.3.3.16.1.3 Context Errors

No requirements.

6.3.3.16.2 Test Suite Files

Test Script: FWK_APT_EDCR_1.scr

Test Applet: FWK_APT_EDCR_1.java

Load Script: FWK_APT_EDCR_1.ldr

Cleanup Script: FWK_APT_EDCR_1.clr

Parameter File: FWK_APT_EDCR_1.par

6.3.3.16.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1	Applet registration to EVENT_EVENT_DOWNLOAD_CARD_READER _STATUS and triggering		
	Applet is registered to the EVENT_EVENT_DOWNLOAD_CARD_READER_STATUS and to EVENT_FORMATTED_SMS_PP_ENV		
	Event=EVENT_EVENT_DOWNLOAD_CARD_READER_STA TUS 1-Toolkit Registry.isEventSet() method is called.		
	2-An Envelope EVENT_DOWNLOAD_CARD_READER_STATUS is sent to the SIM.	2- Applet is triggered	

ld	Description	API/Framework Expectation	APDU Expectation
2	Applet deregistration		
	Event= EVENT_EVENT_DOWNLOAD_CARD_READER_STATUS		
	Toolkit Registry.clearEvent()method is called		
	Perform SIM initialization with all the facilities supported		
	1-An Envelope EVENT_DOWNLOAD_CARD_READER_STATUS is sent to the SIM.	1- Applet isn't triggered	
	An Envelope formatted sms pp envelope is sent to the sim		
	Event= EVENT_EVENT_DOWNLOAD_CARD_READER_STATUS		
	Toolkit Registry.setEvent() method is called Perform SIM initialization with all the facilities supported		
	2-An Envelope EVENT_DOWNLOAD_CARD_READER_STATUS is sent to the SIM.	2- Applet is triggered	

6.3.3.16.4 Test Coverage

CRR Number Test Case Number	
CRRN1	1, 2
CRRN2	2

6.3.3.17 EVENT_UNRECOGNIZED_ENVELOPE

Test Area Reference: FWK_APT_EUEV

6.3.3.17.1 Conformance Requirement

6.3.3.17.1.1 Normal Execution

- CRRN1: The applet is triggered by the EVENT_UNRECOGNIZED_ENVELOPE once it has registered to this event and an Unrecognized Envelope is received.
- CRRN2: The applet is not triggered by the EVENT_UNRECOGNIZED_ENVELOPE once it has deregistered from this event.

6.3.3.17.1.2 Parameters error

No requirements.

6.3.3.17.1.3 Context Errors

No requirements.

6.3.3.17.2 Test Suite Files

Test Script: FWK_APT_EUEN_1.scr

Test Applet: FWK_APT_EUEN_1.java

Load Script: FWK_APT_EUEN_1.ldr

Cleanup Script: FWK_APT_EUEN_1.clr

Parameter File: FWK_APT_EUEN_1.par

6.3.3.17.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1	Applet registration to EVENT_UNRECOGNIZED_ENVELOPE and triggering		
	Applet is registered to the EVENT_UNRECOGNIZED_ENVELOPE and to EVENT_FORMMATTED_SMS_PP_ENV		
	Event= EVENT_UNRECOGNIZED_ENVELOPE 1-Toolkit Registry.isEventSet() method is called.	1- Method returns true	
	2-An Envelope UNRECOGNIZED_ENVELOPE is sent to the SIM.	2- Applet is triggered	
2	Applet deregistration		
	Event= EVENT_UNRECOGNIZED_ENVELOPE Toolkit Registry.clearEvent()method is called		
	1-An Envelope UNRECOGNIZED_ENVELOPE is sent to the SIM.	1- Applet isn't triggered	
	a formatted sms pp envelope is sent to the sim		
	Event= EVENT_UNRECOGNIZED_ENVELOPE Toolkit Registry.setEvent() method is called		
	2-An Envelope UNRECOGNIZED_ENVELOPE is sent to the SIM.	2- Applet is triggered	

6.3.3.17.4 Test Coverage

CRR Number	Number Test Case Number	
CRRN1	1, 2	
CRRN2	2	

6.3.3.18 EVENT_STATUS_COMMAND

 $Test\ Area\ Reference:\ FWK_APT_ESTC$

6.3.3.18.1 Conformance Requirement

6.3.3.18.1.1 Normal Execution

- CRRN1: The applet is triggered by the EVENT_STATUS_COMMAND once it has registered to this event and a Status Command is received.
- CRRN2: The applet is not triggered by the EVENT_STATUS_COMMAND once it has deregistered from this
 event.

6.3.3.18.1.2 Parameters error

No requirements.

6.3.3.18.1.3 Context Errors

No requirements.

6.3.3.18.2 Test Suite Files

Test Script: FWK_APT_ESTC_1.scr

Test Applet: FWK_APT_ESTC_1.java

FWK_APT_ESTC_2.java

FWK_APT_ESTC_3.java

Load Script: FWK_APT_ESTC_1.ldr

Cleanup Script: FWK_APT_ESTC_1.clr

Parameter File: FWK_APT_ESTC_1.par

6.3.3.18.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1	Applets registration to		
	EVENT_STATUS_COMMAND and triggering		
	2007-61		
	Applet1 is registered to EVENT_STATUS_COMMAND using the		
	requestPollInterval() command.		
	requestrollimetral() communa.		
	Applet2 is registered to		
	EVENT_STATUS_COMMAND using the		
	RequestPollInterval() command.		
	2001-62		
	Applet3 is registered to EVENT_FORMATTED_SMS_PP_ENV.		
	EVENT_FORMATTED_SMS_FF_ENV.		
	1-A status command is sent to SIM		
		1- Applet1 is triggered.	
		Applet1 finalizes	
		2- Applet2 is triggered.	
		A 1 10 5 1	
		Applet2 finalizes	
		O Applet O is met trianged	
		3- Applet 3 is not triggered	

ld	Description	API/Framework Expectation	APDU Expectation
2	Applet deregistration and registration of the third applet to EVENT_STATUS_COMMAND. The STF shall not reply busy to a call control envelope 1-A formatted sms pp envelope is sent to SIM	1- Applet3 is triggered.	
	Applet3 builds a DISPLAY TEXT. 2- ProactiveHandler.send() is called		2- A proactive command DISPLAY TEXT is sent and applet is suspended until the terminal response
	3-A status command is sent to SIM.	3- Applet1 is triggered.	
	requestPollInteval with POLL_NO_DURATION is called		
	requestPollInteval with POLL_NO_DURATION is called	Applet1 finalizes 4- Applet2 is triggered.	
	requestPollInterval() method is called.	Applet2 finalizes	
		Applet3 finalizes	5- TERMINAL RESPONSE of DISPLAY TEXT is sent to the SIM
3	Applet3 triggering		
	Perform SIM initialization with all the facilities supported Status command is sent to SIM.	Applet3 is triggered. (Applet1 and Applet2 are not triggered)	

6.3.3.18.4 Test Coverage

CR Number	Test Case Number
CRRN1	1, 2, 3
CRRN2	3

6.3.3.19 EVENT_FORMATTED_SMS_CB

Test Area Reference: FWK_APT_EFCB

6.3.3.19.1 Conformance Requirement

6.3.3.19.1.1 Normal Execution

- CRRN1: The applet is triggered by the EVENT_FORMATTED_SMS_CB once:
 - it has been registered to this event;
 - an envelope APDU carrying a Cell Broadcast Page, formatted according to 3GPP TS 23.048 [8], is received;

- the toolkit applet to be triggered is registered with the corresponding TAR in the CB page;
- the security is verified.
- CRRN2: The applet is not triggered by the EVENT_FORMATTED_SMS_CB once it has deregistered from this event.

6.3.3.19.1.2 Parameters error

No requirements.

6.3.3.19.1.3 Context Errors

No requirements.

6.3.3.19.2 Test Suite Files

Test Script: FWK_APT_EFCB_1.scr

Test Applet: FWK_APT_EFCB_1.java

Load Script: FWK_APT_EFCB_1.ldr

Cleanup Script: FWK_APT_EFCB_1.clr

Parameter File: FWK_APT_EFCB_1.par

6.3.3.19.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1	Applet registration to EVENT_FORMATTED_SMS_CB and triggering Applet is registered to EVENT_FORMATTED_SMS_CB and EVENT_FORMATTED_SMS_PP_ENV		
	1-An Envelope EVENT_FORMATTED_SMS_CB is sent to the SIM.	1-Applet is triggered	
2	Applet deregistration ToolkitRegistry.clearEvent() method is called for EVENT_FORMATTED_SMS_CB 1-A formatted SMS CB envelope is sent to the SIM. 2-An envelope SMS-PP formatted is sent to the SIM ToolkitRegistry.setEvent() method is called for EVENT_FORMATTED_SMS_CB 3-An Envelope FORMATTED_SMS_CB is sent to the SIM	1- Applet is not triggered2- Applet is triggered	
		3- Applet is triggered	

6.3.3.19.4 Test Coverage

C	R Number	Test Case Number
CRRN1 (See note)		1, 2
CRRN2		2
NOTE:	IOTE: The security checks are not relevant to the test designed in this test area; they will be checked subclause 6.3.6.	

6.3.3.20 EVENT_EVENT_DOWNLOAD_LANGUAGE_SELECTION

Test Area Reference: FWK_APT_EDLG

6.3.3.20.1 Conformance Requirement

6.3.3.20.1.1 Normal Execution

- CRRN1: The applet is triggered by the EVENT_EVENT_DOWNLOAD_LANGUAGE_SELECTION once it has registered to this event and an Envelope Event DownLoad Language Selection is received.
- CRRN2: The applet is not triggered by the EVENT_EVENT_DOWNLOAD_LANGUAGE_SELECTION once
 it has deregistered from this event.

6.3.3.20.1.2 Parameters error

No requirements.

6.3.3.20.1.3 Context Errors

No requirements.

6.3.3.20.2 Test Suite Files

Test Script: FWK_APT_EDLG_1.scr

Test Applet: FWK_APT_EDLG_1.java

Load Script: FWK_APT_EDLG_1.ldr

Cleanup Script: FWK_APT_EDLG_1.clr

Parameter File: FWK_APT_EDLG_1.par

6.3.3.20.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1	Applet registration to EVENT_EVENT_DOWNLOAD_LANGUAGE_SE LECTION and triggering		
	Applet is registered to the EVENT_EVENT_DOWNLOAD_LANGUAGE_SELECTION and to EVENT_FORMATTED_SMS_PP_ENV.		
	Event= EVENT_EVENT_DOWNLOAD_LANGUAGE_SELECTION		
	1-Toolkit Registry.isEventSet() method is called.	1-Method returns true	
	2-An Envelope EVENT_DOWNLOAD_LANGUAGE_SELECTION is sent to the SIM.	2- Applet is triggered	

ld	Description	API/Framework Expectation	APDU Expectation
2	Applet deregistration		
	Event= EVENT_EVENT_DOWNLOAD_LANGUAGE_SELECTION Toolkit Registry.clearEvent()method is called		
	Perform SIM initialization with Profile Download, SMS PP Data Download, Command Result and Language Selection facilities.		
	1-An Envelope EVENT_DOWNLOAD_LANGUAGE_SELECTION is sent to the SIM.	1- Applet isn't triggered	
	a formatted sms pp envelope is sent to the sim		
	Event= EVENT_EVENT_DOWNLOAD_LANGUAGE_SELECTION Toolkit Registry.setEvent() method is called		
	Perform SIM initialization with Profile Download, SMS PP Data Download, Command Result and Language Selection facilities.		
	2-An Envelope EVENT_DOWNLOAD_LANGUAGE_SELECTION is sent to the SIM.	2- Applet is triggered	

6.3.3.20.4 Test Coverage

CR Number	Test Case Number
CRRN1	1, 2
CRRN2	2

6.3.3.21 EVENT_EVENT_DOWNLOAD_BROWSER_TERMINATION

Test Area Reference: FWK_APT_EDBT

6.3.3.21.1 Conformance Requirement

6.3.3.21.1.1 Normal Execution

- CRRN1: The applet is triggered by the EVENT_EVENT_DOWNLOAD_BROWSER_TERMINATION once it has registered to this event and an Envelope Event DownLoad Browser Termination is received.
- CRRN2: The applet is not triggered by the EVENT_EVENT_DOWNLOAD_BROWSER_TERMINATION once it has deregistered from this event.

6.3.3.21.1.2 Parameters error

No requirements.

6.3.3.21.1.3 Context Errors

No requirements.

6.3.3.21.2 Test Suite Files

Test Script: FWK_APT_EDBT_1.scr

Test Applet: FWK_APT_EDBT_1.java

Load Script: FWK_APT_EDBT_1.ldr

Cleanup Script: FWK_APT_EDBT_1.clr

Parameter File: FWK_APT_EDBT_1.par

6.3.3.21.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1	Applet registration to EVENT_EVENT_DOWNLOAD_ BROWSER_TERMINATION and triggering		
	Applet is registered to the EVENT_EVENT_DOWNLOAD_BROWSER_TERMINATION and to EVENT_FORMATTED_SMS_PP_ENV		
	Event= EVENT_EVENT_DOWNLOAD_BROWSER_TERMINATION		
	1-Toolkit Registry.isEventSet() method is called.	1-Method returns true	
	2-An Envelope EVENT_DOWNLOAD_BROWSER_TERMINATION is sent to the SIM.	2- Applet is triggered	
2	Applet deregistration		
	Event= EVENT_EVENT_DOWNLOAD_BROWSER_TERMINATION Toolkit Registry.clearEvent()method is called Perform SIM initialization with Profile Download, SMS PP Data Download, Command Result and Browser Termination facilities.		
	1-An Envelope EVENT_DOWNLOAD_BROWSER_TERMINATION is sent to the SIM.	1- Applet isn't triggered	
	a formatted sms pp envelope is sent to the sim		
	Event= EVENT_EVENT_DOWNLOAD_BROWSER_TERMINATION Toolkit Registry.setEvent() method is called		
	Perform SIM initialization with Profile Download, SMS PP Data Download, Command Result and Browser Termination facilities.		
	2-An Envelope EVENT_DOWNLOAD_BROWSER_TERMINATION is sent to the SIM.	2- Applet is triggered	

6.3.3.21.4 Test Coverage

CR Number	Test Case Number	
CRRN1	1, 2	
CRRN2	2	

6.3.3.22 EVENT_FIRST_COMMAND_AFTER_SELECT

Test Area Reference: FWK_APT_EFCA

6.3.3.22.1 Conformance Requirement

6.3.3.22.1.1 Normal Execution

- CRRN1: The applet is triggered by the EVENT_FIRST_COMMAND_AFTER_SELECT once it has registered to this event; Upon reception of the first command received by the GSM application after it has been selected, or after the ATR if it is the default application, and before the Status Word of the processed command has been sent back by the GSM application, the toolkit framework shall trigger all the toolkit applets registered to this event.
- CRRN2: The applet is not triggered by the EVENT_FIRST_COMMAND_AFTER_SELECT once it has deregistered from this event.
- CRRN3: If the first command received by the GSM application is a toolkit applet triggering command (e.g.
 TERMINAL PROFILE), the toolkit applets registered on the EVENT_FIRST_COMMAND_AFTER_SELECT
 event shall be triggered first.

6.3.3.22.2 Test Suite Files

Test Script: FWK_APT_EFCA_1.scr

Test Applet: FWK_APT_EFCA_1.java

FWK_APT_EFCA_2.java

FWK_APT_EFCA_3.java

FWK_APT_EFCA_4.java

FWK_APT_EFCA_5.java

Load Script: FWK_APT_EFCA_1.ldr

Cleanup Script: FWK_APT_EFCA_1.clr

Parameter File: FWK_APT_EFCA_1.par

6.3.3.22.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1	Applets registration to EVENT_FIRST_COMMAND_AFTER_SELECT and triggering		
	Applet1 is registered to the EVENT_FIRST_COMMAND_AFTER_SELECT		
	Applet2 is registered to the EVENT_PROFILE_DOWNLOAD.		
	Applet3 is registered to EVENT_FORMATTED_SMS_PP_ENV.		
	1-Terminal Profile command is sent to the SIM. Applet1 deregisters from EVENT_FIRST_COMMAND_AFTER_SELECT.	1- Applet1 is triggered by EVENT_FIRST_COMMAND_AFTE R_SELECT	
	2- Applet2 deregisters from EVENT_PROFILE_DOWNLOAD.	Applet1 finalizes Applet2 is triggered by EVENT_PROFILE_DOWNLOAD	
	3-Envelope(SMS-PP-DOWNLOAD) formatted is	Applet2 finalizes Applet3 is not triggered	
	sent to the SIM	3-Applet3 is triggered.	
	4-Applet3 calls setEvent() on event EVENT_FIRST_COMMAND_AFTER_SELECT.		
2	Deregistered applets are not triggered	1-Applet3 is triggered. Applet1 and Applet2 are not triggered.	
	1-Reset then Terminal Profile command is sent to the SIM		
	2-Applet3 calls setEvent() on EVENT_PROFILE_DOWNLOAD.	2-Applet3 finalizes.	
3	Install a 4 th applet registered to EVENT_FIRST_COMMAND_AFTER_SELECT and EVENT_PROFILE_DOWNLOAD Applet4 is installed, with the same	1- Applet4 is triggered by EVENT_FIRST_COMMAND_AFTE R_SELECT.	
	priority level as Applet3. 1-Reset then Terminal Profile command is sent to the SIM	Applet3 is triggered by EVENT_FIRST_COMMAND_AFTE R_SELECT.	
		Applet4 is triggered by EVENT_PROFILE DOWNLOAD.	
	Delete all applets.	Applet3 is triggered by EVENT_PROFILE_DOWNLOAD.	
4	Check that the applet is triggered before the first SW is sent. 1-Install Applet 5. Applet 5 is registered with two entries in the menu entries list. Applet5 is also registered to		3-The SETUP MENU proactive command is fetched. There is only one item for Applet5.
	EVENT_FIRST_COMMAND_AFTER_SELECT. 2-Reset and TERMINAL PROFILE.	2- Applet 5 is triggered	
	3-Applet disables a menu entry.		

NOTE: Testing the triggering of an applet upon the first command after select is not possible.

6.3.3.22.4 Test Coverage

CR Number	Test Case Number
CRRN1	1,2,3, 4
CRRN2	3
CRRN3	1, 4

6.3.3.23 EVENT_EVENT_DOWNLOAD_DATA_AVAILABLE

Test Area Reference: FWK_APT_EDDA

6.3.3.23.1 Conformance Requirement

6.3.3.23.1.1 Normal Execution

- CRRN1: For EVENT_EVENT_DOWNLOAD_DATA_AVAILABLE, the framework shall only trigger the applet registered to this event with the appropriate channel identifier.
- CRRN2: The registration to the EVENT_EVENT_DOWNLOAD_DATA_AVAILABLE is effective once the toolkit applet has issued a successful OPEN CHANNEL proactive command, and valid till the first successful CLOSE CHANNEL or the end of card session.
- CRRN3: When a Toolkit Applet has sent an OPEN CHANNEL proactive command and received a successful TERMINAL RESPONSE, the framework shall register the received channel identifier for the calling Toolkit Applet.
- CRRN4: When a Toolkit Applet has sent a CLOSE CHANNEL proactive command and received a successful TERMINAL RESPONSE, the framework shall release the channel identifier contained in the command. A successful TERMINAL RESPONSE means that the result of the proactive command execution belongs to command performed category (i.e. General Result ='0x').

6.3.3.23.2 Test Suite Files

Test Script: FWK_APT_EDDA_1.scr

Test Applet: FWK_APT_EDDA_1.java

Load Script: FWK_APT_EDDA_1.ldr

Cleanup Script: FWK_APT_EDDA_1.clr

Parameter File: FWK_APT_EDDA_1.par

6.3.3.23.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1	Applet registration to		
	EVENT_EVENT_DOWNLOAD_DATA_AVAILAB	1- Applet1 is triggered by	
	2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	Unformatted SMS PP envelope.	
	Applet1 is registered to Unformatted SMS PP Envelope.		
	1- Unformatted SMS PP envelope is sent to the SIM.		
	2- Applet calls setEvent() with the event		
	EVENT_EVENT_DOWNLOAD_DATA_AVAILABLE.	2- Applet1 finalizes.	
	3- An envelope Event Download Data		
	Available is sent to the SIM Channel Status = 81 00	3- Applet1 is not triggered.	
	4- Unformatted SMS PP envelope is sent to	4. Applet4 is triangual by	
	the SIM.	4- Applet1 is triggered by Unformatted SMS PP envelope.	
	5- Applet1 builds a proactive command OPEN CHANNEL calling ProactiveHandler.init() method.		
			6- OPEN CHANNEL
	6- send() method is called to register to this event.		proactive command is fetched.
		7- Applet1 finalizes.	Unsuccessful TERMINAL RESPONSE of OPEN
	8- An envelope Event Download Data Available is sent to the SIM with Channel Status = 01 00.	8- Applet1 is not triggered.	CHANNEL is sent to the SIM.
		9- Applet1 is triggered by	
	9- Unformatted SMS PP envelope is sent to the SIM.	EVENT_UNFORMATTED_SMS_P P_ENV.	
	10- Applet1 builds a proactive command OPEN CHANNEL calling ProactiveHandler.init() method.		
	11- send() method is called to register to		
	this event.	12- Applet1 finalizes.	11- OPEN CHANNEL
			proactive command is
			fetched.
			Successful TERMINAL RESPONSE of OPEN
			CHANNEL is sent to the
2	Applet triggering to		SIM with Channel Id = 01.
	EVENT_EVENT_DOWNLOAD_DATA_AVAILAB LE		
	1- An envelope Event Download Data		
	Available is sent to the SIM Channel Status = 81 00.		
		1- Applet1 is triggered.	
3	Applet deregistration to EVENT_EVENT_ DOWNLOAD_DATA_ AVAILABLE		
	0- Unformatted SMS PP envelope is sent to the SIM.	0- Applet1 is triggered.	1- OPEN CHANNEL proactive command is fetched.
	1- Applet1 initialises and sends an OPEN CHANNEL proactive command.		Successful terminal response is sent, with channelld=02.
	2- Applet1 builds a CLOSE CHANNEL		

Description	API/Framework Expectation	APDU Expectation
Proactive Command calling ProactiveHandler.initCloseChannel() and ProactiveHandler.send() methods. 3- An envelope Event Download Data Available is sent to the SIM. Channel Status = 82 00	3- Applet1 is triggered.	2- CLOSE CHANNEL proactive command is fetched. Unsuccessful TERMINAL RESPONSE of CLOSE CHANNEL is sent to the SIM.
4- Applet1 builds a CLOSE CHANNEL Proactive Command calling ProactiveHandler.initCloseChannel() and ProactiveHandler.send() methods.	5- Applet1 finalizes.	4- CLOSE CHANNEL proactive command is fetched. Successful TERMINAL RESPONSE of CLOSE CHANNEL is sent to the SIM with Channel Id = 02.
Applet triggering to EVENT_EVENT_DOWNLOAD_DATA_AVAILAB LE 1- An envelope Event Download Data Available is sent to the SIM Channel Status = 82 00.	1- Applet1 is not triggered.	
Applet1 not triggered after a reset O- Applet1 is triggered by an unformatted		
1- Applet1 builds a proactive command OPEN CHANNEL calling ProactiveHandler.init() method.	3- returns true.	1- OPEN CHANNEL proactive command is fetched.
<pre>2- send() method is called to register to this event. 3- isEventSet() method is called.</pre>		2- Successful TERMINAL RESPONSE of OPEN CHANNEL is sent to the
4- Reset the card. 5- An envelope Event Download Data Available is sent to the SIM Channel Status = 82 00.	5- Applet1 is not triggered.	SIM with Channel Id = 02.
	ProactiveHandler.initCloseChannel() and ProactiveHandler.send() methods. 3- An envelope Event Download Data Available is sent to the SIM. Channel Status = 82 00 4- Applet1 builds a CLOSE CHANNEL Proactive Command calling ProactiveHandler.initCloseChannel() and ProactiveHandler.send() methods. Applet triggering to EVENT_EVENT_DOWNLOAD_DATA_AVAILAB LE 1- An envelope Event Download Data Available is sent to the SIM Channel Status = 82 00. Applet1 not triggered after a reset 0- Applet1 is triggered by an unformatted SMS PP Envelope 1- Applet1 builds a proactive command OPEN CHANNEL calling ProactiveHandler.init() method. 2- send() method is called to register to this event. 3- isEventSet() method is called. 4- Reset the card. 5- An envelope Event Download Data Available is sent to the SIM	Proactive Command calling ProactiveHandler.initCloseChannel() and ProactiveHandler.send() methods. 3- An envelope Event Download Data Available is sent to the SIM. Channel Status = 82 00 5- Applet1 finalizes. 4- Applet1 builds a CLOSE CHANNEL Proactive Command calling ProactiveHandler.initCloseChannel() and ProactiveHandler.send() methods. Applet triggering to EVENT_EVENT_DOWNLOAD_DATA_AVAILAB LE 1- An envelope Event Download Data Available is sent to the SIM Channel Status = 82 00. Applet1 not triggered after a reset 0- Applet1 is triggered by an unformatted SMS PP Envelope 1- Applet1 builds a proactive command OPEN CHANNEL calling ProactiveHandler.init() method. 2- send() method is called to register to this event. 3- isEventSet() method is called. 4- Reset the card. 5- An envelope Event Download Data Available is sent to the SIM 5- Applet1 is not triggered.

6.3.3.23.4 Test Coverage

CR Number	Test Case Number
CRRN1	2
CRRN2	1, 4, 5
CRRN3	1
CRRN4	3

6.3.3.24 EVENT_EVENT_DOWNLOAD_CHANNEL_STATUS

Test Area Reference: FWK_APT_EDCS

6.3.3.24.1 Conformance Requirement

6.3.3.24.1.1 Normal Execution

• CRRN1: For EVENT_EVENT_DOWNLOAD_CHANNEL_STATUS, the framework shall only trigger the applet registered to this event with the appropriate channel identifier.

- CRRN2: The registration to the EVENT_EVENT_DOWNLOAD_CHANNEL_STATUS is effective once the toolkit applet has issued a successful OPEN CHANNEL proactive command, and valid till the first successful CLOSE CHANNEL or the end of the card session.
- CRRN3: When a Toolkit Applet has sent an OPEN CHANNEL proactive command and received a successful TERMINAL RESPONSE, the framework shall register the received channel identifier for the calling Toolkit Applet.
- CRRN4: When a Toolkit Applet has sent a CLOSE CHANNEL proactive command and received a successful TERMINAL RESPONSE, the framework shall release the channel identifier contained in the command. A successful TERMINAL RESPONSE means that the result of the proactive command execution belongs to command performed category (i.e. General Result ='0x').

6.3.3.24.2 Test Suite Files

Test Script: FWK APT EDCS 1.scr

Test Applet: FWK_APT_EDCS_1.java

Load Script: FWK_APT_EDCS_1.ldr

Cleanup Script: FWK_APT_EDCS_1.clr

Parameter File: FWK_APT_EDCS_1.par

6.3.3.24.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1	Applet registration to		
	EVENT_EVENT_DOWNLOAD_CHANNEL_STAT US	Unformatted SMS PP envelope	
	Applet1 is registered to Unformatted SMS PP Envelope. 1-Unformatted SMS PP envelope is sent to the SIM.	2- Applet1 finalizes.	6- OPEN CHANNEL proactive command is fetched. Unsuccessful TERMINAL RESPONSE of OPEN CHANNEL is sent to the
	2-The applet calls setEvent() with EVENT_EVENT_DOWNLOAD_CHANNEL_STATUS.		SIM. 11- OPEN CHANNEL
	3- An envelope Event Download Channel Status is sent to the SIM. Channel Status = 81 00	3- Applet1 is not triggered.	proactive command is fetched. Successful TERMINAL
	4-Unformatted SMS PP envelope is sent to the SIM.	4- Applet1 is triggered by Unformatted SMS PP envelope.	RESPONSE of OPEN CHANNEL is sent to the SIM with Channel Id = 01.
	5- Applet1 builds a proactive command OPEN CHANNEL calling ProactiveHandler.init() method.		
	6- send() method is called to register to this event.	7- Applet finalizes.	
	8- An envelope Event Download Data Available is sent to the SIM with Channel Status = 01 00.	8- Applet1 is not triggered.	
	9- Unformatted SMS PP envelope is sent to the SIM.		
	10- Applet1 builds a proactive command OPEN CHANNEL calling ProactiveHandler.init() method.	9- Applet1 is triggered by EVENT_UNFORMATTED_SMS_P P_ENV.	
	11- send() method is called to register to this event a second time.		
		12- Applet1 finalizes.	
2	Applet triggering to EVENT_EVENT_DOWNLOAD_CHANNEL STATUS		
	1- An envelope Event Download Channel Status is sent to the SIM. Channel Status = 81 00	1- Applet1 is triggered.	
3	Applet deregistration to EVENT_EVENT_		
	DOWNLOAD_CHANNEL STATUS 0- Unformatted SMS PP envelope is sent to the SIM. 1-Applet1 initialises and sends an OPEN	0- Applet1 is triggered.	OPEN CHANNEL proactive command is fetched. Successful terminal response is sent, with
	CHANNEL proactive command.		channelld=02.
	2- Applet1 builds a CLOSE CHANNEL Proactive Command calling ProactiveHandler.initCloseChannel() and ProactiveHandler.send() methods.	3- The applet is triggered.	2-CLOSE CHANNEL proactive command is fetched. Unsuccessful TERMINAL

ld	Description	API/Framework Expectation	APDU Expectation
	3-An envelope Event Download Channel Status is sent to the SIM. Channel Status = 82 00	5- Applet1 finalizes.	RESPONSE of CLOSE CHANNEL is sent to the SIM. 4- CLOSE CHANNEL proactive command is
	4- Applet1 builds a Close Channel Proactive Command calling ProactiveHandler.initCloseChannel() and ProactiveHandler.send() methods.		fetched. Successful TERMINAL RESPONSE of CLOSE CHANNEL is sent to the SIM with Channel Id = 02.
4	Applet triggering to EVENT_EVENT_DOWNLOAD_CHANNEL STATUS		
	1- An envelope Event Download Channel Status is sent to the SIM. Channel Status = 82 00	Applet1 is not triggered.	
5	Applet1 not triggered after a reset		
	0- Applet1 is triggered by an unformatted SMS PP Envelope.		
	1- Applet1 builds a proactive command OPEN CHANNEL calling ProactiveHandler.init() method.	3- returns true.	1- OPEN CHANNEL proactive command is fetched.
	2- send() method is called to register to this event.		2- Successful TERMINAL
	3- isEventSet() method is called.		RESPONSE of OPEN CHANNEL is sent to the
	4- Reset the card.	5- Applet1 is not triggered.	SIM with Channel Id = 02.
	5- An envelope Event Download Data Available is sent to the SIM Channel Status = 82 00.		

6.3.3.24.4 Test Coverage

CR Number	Test Case Number
CRRN1	2
CRRN2	1, 4, 5
CRRN3	1
CRRN4	3

6.3.3.25 EVENT_FORMATTED_SMS_PP_UPD

Test Area Reference: FWK_APT_EFSU

6.3.3.25.1 Conformance Requirement

6.3.3.25.1.1 Normal Execution

- CRRN1: The applet is triggered by the EVENT_FORMATTED_SMS_PP_UPD once:
 - it has been registered to this event,
 - a Short Message Point to Point (Single or Concatenated) is received by Update Record EFsms APDU(s) and is formatted according to TS 23.048 [8],
 - the toolkit applet to be triggered is registered with the corresponding TAR in the SMS TPDU,

• CRRN2: The applets are not triggered by the EVENT_FORMATTED_SMS_PP_UPD once it has deregistered from this event.

6.3.3.25.2 Test Suite Files

Test Script: FWK_APT_EFSU_1.scr

Test Applet: FWK_APT_EFSU_1.java

Load Script: FWK_APT_EFSU_1.ldr

Cleanup Script: FWK_APT_EFSU_1.clr

Parameter File: FWK_APT_EFSU_1.par

6.3.3.25.3 Test Procedure

6.3.3.25.4 Test Coverage

ld	Desc	cription	API/Framework Expectation	APDU Expectation
1	Applet regist	ration to EVENT PP_UPD and triggering		
	Applet is registered EVENT_FORMATTED_SMS_ EVENT_UNRECOGNIZED_E	PP_UPD and		
	1. Toolkit Registry.icalled for EVENT_FOR	sEventSet() method is MATTED_SMS_PP_UPD	1- The method returns true.	
	2. Short Message Poi Formatted is receive EFsms APDU.		2- Applet is triggered.	
	3. Short Message Poi Concatenated Formatt Update Record EFsms Concatenated Message Short Messages. The Short Message is 70 70).	ed is received by APDU(s) (The is composed of 2 UDL for the first	3- Applet is triggered on reception of the last concatenated SMS	
2	Applet de	registration		
	ToolkitRegistry.cleacalled for EVENT_FOR			
	1. Short Message Poi Formatted is receive EFsms APDU.	nt to Point Single and d by Update Record	1- Applet is not triggered	
	2. Short Message Poi Concatenated and For Update Record EFsms Concatenated Message Short Messages. The Short Message is 70 70).	matted is received by APDU(s). (The is composed of 2 UDL for the first	2- Applet is not triggered	
	An unrecognized enve	lope is sent to the		
	ToolkitRegistry.setE called for EVENT_FOR		3- Applet is triggered	
	3. Short Message Poi Formatted is receive EFsms APDU.	nt to Point Single and d by Update Record		
	4. Short Message Poi Concatenated Formatt Update Record EFsms Concatenated Message Short Messages. The Short Message is 70	ed is received by APDU(s). (The is composed of 2 UDL for the first	4- Applet is triggered on reception of the last concatenated SMS.	
	70).			

CRR Number	rest case number
CRRN1 (See note)	1,2
CRRN2	2

NOTE: The security checks are not relevant to the test designed in this test area; they will be checked in the "Framework Security Management" section.

6.3.3.26 EVENT_UNFORMATTED_SMS_PP_UPD

Test Area Reference: FWK_APT_EUSU

6.3.3.26.1 Conformance Requirement

6.3.3.26.1.1 Normal Execution

- CRRN1: The applets registers are triggered by the EVENT_UNFORMATTED_SMS_PP_UPD once a Short Message Point to Point (Single or Concatenated) is received by Update Record EFsms APDU(s) and is unformatted.
- CRRN2: The applets are not triggered by the EVENT_UNFORMATTED_SMS_PP_UPD once it has deregistered from this event.

6.3.3.26.2 Test Suite Files

Test Script: FWK_APT_EUSU_1.scr

Test Applet: FWK_APT_EUSU_1.java

Load Script: FWK_APT_EUSU_1.ldr

Cleanup Script: FWK_APT_EUSU_1.clr

Parameter File: FWK_APT_EUSU_1.par

6.3.3.26.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1	Applet registration to EVENT	-	
	UNFORMATTED_SMS_PP_UPD and triggering		
	Applet is registered to EVENT_UNFORMATTED_SMS_PP_UPD and EVENT_UNRECOGNIZED_ENVELOPE		
	1. Toolkit Registry.isEventSet() method is called for EVENT_UNFORMATTED_SMS_PP_UPD	1- Applet is not triggered	
	2. Short Message Point to Point Single and Unformatted is received by Update Record EFsms APDU	2- Applet is triggered.	
	3. Short Message Point to Point Concatenated and Unformatted is received by Update Record EFsms APDU (The Concatenated Message is composed of 2 Short Messages. The UDL for the first Short Message is 70 and for the second 70).	3- Applet is triggered on reception of the last concatenated SMS.	
2	Applet deregistration		
	ToolkitRegistry.clearEvent() method is called for EVENT_UNFORMATTED_SMS_PP_UPD		
	1. Short Message Point to Point Single and Unformatted is received by Update Record EFsms APDU		
	2. Short Message Point to Point	- Applet is not triggered	
	Concatenated and Unformatted is received by Update Record EFsms APDU(s) (The Concatenated Message is composed of 2 Short Messages. The UDL for the first Short Message is 70 and for the second 70).	2- Applet is not triggered.	
	An unrecognized envelope is sent to the sim		
	ToolkitRegistry.setEvent() method is called for EVENT_UNFORMATTED_SMS_PP_UPD	3- Applet is triggered	
	3. Short Message Point to Point Single and Unformatted is received by Update Record EFsms APDU		
	4. Short Message Point to Point Concatenated and Unformatted is received by Update Record EFsms APDU(s) (The Concatenated Message is composed of 2 Short Messages. The UDL for the first Short Message is 70 and for the second 70).	4- Applet is triggered on reception of the last concatenated SMS	

6.3.3.26.4 Test Coverage

CRR Number	Test Case Number
CRRN1	1,2
CRRN2	2

6.3.4 Proactive Command Sending by the STF

6.3.4.1 System Proactive Commands

Test Area Reference: FWK_PCS_SPCO

6.3.4.1.1 Conformance Requirements

6.3.4.1.1.1 Normal Execution

- CRRN1: When a toolkit applet changes a menu entry of its registry object, the SIM Toolkit Framework shall dynamically* update the menu stored in the ME during the current card session
- CRRN2: The STF shall use the data of the EFsume file when issuing the SET UP MENU proactive command.
- CRRN3: For all EVENT_EVENT_DOWNLOAD_*: When a toolkit applet changes one or more of these requested events of its registry object, the STF shall dynamically* update the event list stored in the ME during the current card session by SET UP EVENT LIST proactive command.

NOTE: *The STF shall send its system proactive command as soon as no proactive session is pending and all the applets registered to the current events have been triggered and have returned from the processToolkit method invocation.

6.3.4.1.1.2 Parameters error

No requirements.

6.3.4.1.1.3 Context Errors

No requirements.

6.3.4.1.2 Test Suite Files

Test Script: FWK_PCS_SPCO_1.scr

Test Applet: FWK_PCS_SPCO_1.java

Load Script: FWK_PCS_SPCO_1.ldr

Cleanup Script: FWK_PCS_SPCO_1.clr

Parameter File: FWK_PCS_SPCO_1.par

6.3.4.1.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1	Install Applet 1, Registered to the EVENT_EVENT_DOWNLOAD_MT_CALL and EVENT_EVENT_DOWNLOAD_ LOCATION_STATUS		setEventList proactive command [Event list]= '19020003' or '99020003'
	Perform SIM initialization with EVENT DOWNLOAD facilities supported		

ld	Description	API/Framework Expectation	APDU Expectation
2	Trigger the applet by ENVELOPE		1. DISPLAY TEXT
	(SMS_FORMATTED_PP) command Clear the events and build a display text		Proactive command
	command		2. SET UP EVENT LIST Proactive command
			[CommandQualifier]= 00h

6.3.4.1.4 Test Coverage

CRR number	Test case number
N1	see:
	subclause6.2.9.2, CRRN1,
	subclause 6.2.9.4, CRRN3,
	subclause 6.2.9.5 CRRN4,
	subclause 6.2.9.8 CRRN1
N2	see:
	subclause 6.2.9.2 CRRN1,
	subclause 6.2.9.8 CRRN1
N3	1,2

6.3.4.2 Interaction with GSM commands

Test Area Reference: FWK_PCS_IGCO

6.3.4.2.1 Conformance Requirements

6.3.4.2.1.1 Normal Execution

• CRRN1: The STF shall process a GSM command even when a proactive command is pending (before and after the FETCH command until the terminal response). The STF shall answer with the SW1 and SW2 described in 3GPP TS 51.011 [3] and 3GPP TS 51.014 [4].

6.3.4.2.1.2 Parameters error

No requirements.

6.3.4.2.1.3 Context Errors

No requirements.

6.3.4.2.2 Test Suite Files

Test Script: FWK_PCS_IGCO_1.scr

Test Applet: FWK_PCS_IGCO_1.java

Load Script: FWK_PCS_IGCO_1.ldr

Cleanup Script: FWK_PCS_IGCO_1.clr

Parameter File: FWK_PCS_IGCO_1.par

6.3.4.2.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1	Interaction with GSM Commands after TERMINAL PROFILE in connection with FETCH and TERMINAL RESPONSE	·	
	Applet is registered to Menu Selection		
	RST TERMINAL PROFILE (Profile: supports all facilities except: SET UP EVENT LIST, POLL INTERVAL and POLLING OFF) 1- System issues a proactive command SETUP_MENU		1- 91xx
	2- SELECT MF 3- GET RESPONSE (6 Bytes) 4- Failed SELECT File 5- FETCH		2- 9Fxx 3- 91xx 4- 9404
	6- SELECT MF 7- GET RESPONSE (6 Bytes) 8- TERMINAL RESPONSE		5- Proactive Command: SETUP MENU
			6- 9Fxx 7- 9000 8- 9000
2	Interaction with GSM Commands after ENVELOPE (MENU SELECTION) in connection with FETCH and TERMINAL RESPONSE		
	Menu Entry ID = 0x01		4 0577
	1- SELECT MF 2- GET RESPONSE (6 Bytes) 3- Failed SELECT File 4- FETCH		1- 9FXX 2- 91XX 3- 9404 4- Proactive Command: DISPLAY TEXT
	5- SELECT MF 6- GET RESPONSE (6 Bytes) 7- TERMINAL RESPONSE		5- 9FXX 6- 9000 7- 9000
3	Interaction with GSM Commands after TERMINAL RESPONSE in proactive command session in connection with FETCH and TERMINAL RESPONSE		
	Menu Entry ID = 0x02		
	1- SELECT MF 2- GET RESPONSE (6 Bytes) 3- FETCH		1- 9FXX 2- 91XX 3- Proactive Command: DISPLAY TEXT
	4- SELECT MF 5- GET RESPONSE (6 Bytes) 6- Failed SELECT File 7- TERMINAL RESPONSE		4- 9FXX 5- 9000 6- 9404 7- 9000
	8- SELECT MF 9- GET RESPONSE (6 Bytes) 10-Failed SELECT File 11-FETCH 12-SELECT MF		8- 9FXX 9- 91XX 10-9404 11-Proactive Command: DISPLAY TEXT
	13-GET RESPONSE (6 Bytes) 14-TERMINAL RESPONSE		12-9FXX 13-9000 14-9000

6.3.4.2.4 Test Coverage

CRR number	Test case number
N1	1, 2, 3

6.3.4.3 Proactive Command Control

Test Area Reference: FWK_PCS_PCCO

6.3.4.1.1 Conformance Requirements

6.3.4.1.1.1 Normal Execution

- CRRN1: The SIM Toolkit Framework shall prevent the toolkit applet to issue the following proactive commands: SET UP MENU, SET UP EVENT LIST, POLL INTERVAL, POLLING OFF. If an applet attempts to issue such a command, the SIM Toolkit Framework shall throw an exception.
- CRRN2: The SIM Toolkit Framework shall prevent a toolkit applet to issue a TIMER MANAGEMENT proactive command using a timer identifier, which is not allocated to it. If an applet attempts to issue such a command, the SIM Toolkit Framework shall throw an exception.
- CRRN3: The SIM Toolkit Framework shall prevent a toolkit applet to issue a SEND DATA, RECEIVE DATA and CLOSE CHANNEL proactive commands using a channel identifier, which is not allocated to it. If an applet attempts to issue such a command the SIM Toolkit Framework shall throw an exception.
- CRRN4: The SIM Toolkit Framework shall prevent a toolkit applet to issue an OPEN CHANNEL proactive command if it exceeds the maximum number of channel allocated to this applet. If an applet attempts to issue such a command the SIM Toolkit Framework shall throw an exception.

6.3.4.1.2 Test Suite Files

Test Script: FWK_PCS_PCCO_1.scr

Test Applet: FWK_PCS_PCCO_1.java

FWK_PCS_PCCO_2.java

FWK_PCS_PCCO_3.java

Load Script: FWK_PCS_PCCO_1.ldr

Cleanup Script: FWK_PCS_PCCO_1.clr

Parameter File: FWK_PCS_PCCO_1.par

6.3.4.1.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
0	Applets installation		
	Applet1 is installed with 4 timers maximum, 0 channel maximum and 1 menu. Applet2 is installed with 8 timers maximum, 3 channels maximum. Applet3 is installed with 1 channel maximum.		
1	STK Proactive Commands		
	1- Send a formatted envelope with the TAR of Applet1 2- Applet1 builds and sends a SET UP MENU proactive command	1- Applet1 is triggered 2- COMMAND_NOT_ALLOWED toolkit exception is thrown	1- 90 00 (no proactive command is sent)

ld	Description	API/Framework Expectation	APDU Expectation
	3- Applet1 builds and sends a SET UP EVENT	3- COMMAND_NOT_ALLOWED	
	LIST proactive command	toolkit exception is thrown	
	4- Applet1 builds and sends a POLL INTERVAL proactive command	4- COMMAND_NOT_ALLOWED toolkit exception is thrown	
	5- Applet1 builds and sends a POLLING OFF	5- COMMAND_NOT_ALLOWED	
	proactive command	toolkit exception is thrown	
2	TIMER MANAGEMENT Proactive command	•	
	1- Send a formatted envelope with the TAR of	1- Applet2 is triggered	
	Applet2 2- Applet2 allocates 8 timers by calling	2- No exception is thrown	
	allocateTimer() method and release the 3 timers from id 1 to 3.	2 No oxoophon to unown	
	3- Send a formatted envelope with the TAR of Applet1	3- Applet1 is triggered	
	4- Applet1 allocates 3 timers (Id 1 to 3) by calling allocateTimer() method 3 times	4- No exception is thrown	
	5- Send a formatted envelope with the TAR of Applet2	5- Applet2 is triggered	
	6- Applet2 releases timers of Id 4 to 7 7- Send a formatted envelope with the TAR of Applet1	6- No exception is thrown 7- Applet1 is triggered	
	8- For each of the 3 timers allocated by Applet1 (Id 1to 3) a TIMER MANAGEMENT proactive session	8- No exception is thrown	
	is performed 9- For other timers (Id 4 to 8), Applet1 builds and sends a TIMER MANAGEMENT proactive command	9- COMMAND_NOT_ALLOWED toolkit exception is thrown	
			O O TIMED MANIACEMENT
			8- 3 TIMER MANAGEMENT proactive commands are
			fetched
			9- The Status word of the
			last previous Terminal
			Response is 90 00 (no more proactive command is sent)
3	No Channel allowed		
	1- Send a formatted envelope with the TAR of	1- Applet1 is triggered	1- 90 00 (no proactive
	Applet1 2- Applet1 builds and sends a CSD OPEN	2- COMMAND_NOT_ALLOWED	command is sent)
		toolkit exception is thrown	
	3- Applet1 builds and sends a GPRS OPEN	3- COMMAND_NOT_ALLOWED	
	CHANNEL proactive command	toolkit exception is thrown	
	4Applet1 builds and sends a SEND DATA	4- COMMAND_NOT_ALLOWED	
	proactive command 5- Applet1 builds and sends a RECEIVE DATA	toolkit exception is thrown 5- COMMAND_NOT_ALLOWED	
	proactive command	toolkit exception is thrown	
		6- COMMAND_NOT_ALLOWED	
A	proactive command 4 Channels allowed	toolkit exception is thrown	
4	4 Channels allowed		
	1- Send a formatted envelope with the TAR of Applet3	1- Applet3 is triggered	
	2- Applet3 builds and sends a CSD OPEN CHANNEL proactive command	2- No exception is thrown	2- 91 1C
	3- Send a Fetch and Terminal Response OK on channel 7		3- OPEN CHANNEL proactive
	4- Send a formatted envelope with the TAR of Applet2	4- Applet2 is triggered	
	5- Applet2 builds and sends a CSD OPEN CHANNEL proactive command	5- No exception is thrown	5- 91 1C
	6- Send a Fetch and Terminal Response OK on channel 1		6- OPEN CHANNEL proactive command is fetched

CH 8- ch	HANNEL proactive command	7- No exception is thrown	7- 91 17
	Send Fetch and Terminal Response OK on nannel 2		8- OPEN CHANNEL proactive command is
			fetched, SW = 91 1C on the Terminal Response
	, · · · ·	9- COMMAND_NOT_ALLOWED toolkit exception is thrown	
10	0- For each channel id from 3 to 7, Applet2 builds	10- COMMAND_NOT_ALLOWED	
	nd sends a RECEIVE DATA proactive command 1- For each channel id from 3 to 7, Applet2 builds	toolkit exception is thrown	
an		toolkit exception is thrown	
	2- Applet2 builds and sends a CSD OPEN HANNEL proactive command	12- No exception is thrown	
13	3- Fetch and Terminal Response OK on channel		13- OPEN CHANNEL proactive command is fetched
	4- Applet2 builds and sends an OPEN CHANNEL roactive command	toolkit exception is thrown	14- 90 00 expected to the previous Terminal Response (no proactive command is sent)

6.3.4.1.4 Test Coverage

CRR number	Test case number
N1	1
N2	2
N3	3,4
N4	3,4

6.3.5 Exception Handling

6.3.5.1 Hide Exceptions from the ME

Test Area Reference: FWK_EXH_HEME

6.3.5.1.1 Conformance Requirements

6.3.5.1.1.1 Normal Execution

CRRN1: A toolkit applet may throw an exception, but this error will not be sent to the ME.

NOTE: Because the behaviour of the SIM is not exactly defined for the above CRRN, there are no tests defined here yet.

6.3.5.1.1.2 Parameters error

No requirements.

6.3.5.1.1.3 Context Errors

No requirements.

6.3.5.2 Interaction with Multiple Triggering

Test Area Reference: FWK_EXH_IMTG

6.3.5.2.1 Conformance Requirements

6.3.5.2.1.1 Normal Execution:

• CRRN1: An exception thrown by a toolkit applet, will not influence toolkit applets registered to the same event.

6.3.5.2.1.2 Parameters error

No requirements.

6.3.5.2.1.3 Context Errors

No requirements.

6.3.5.2.2 Test Suite Files

Test Script: FWK_EXH_IMTG_1.scr

Test Applet: FWK_EXH_IMTG_1.java

Load Script: FWK_EXH_IMTG_1.ldr

Cleanup Script: FWK_EXH_IMTG_1.clr

Parameter File: FWK_EXH_IMTG_1.par

6.3.5.2.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
0	Load/install 2 toolkit applets registered to EVENT_STATUS_COMMAND, EVENT_PROFILE_DOWNLOAD, EVENT_UNRECOGNIZED_ENVELOPE, EVENT_EVENT_DOWNLOAD_MT_CALL, EVENT_UNFORMATTED_SMS_PP_ENV, EVENT_UNFORMATTED_SMS_PP_UPD, EVENT_UNFORMATTED_SMS_CB applet1: Priority= 0x01, applet2: Priority= 0x02,		
	(i.e. applet1 is triggered before applet2)		
1	Status_Command is sent		
		1- Applet1 is triggered	
		2- NullPointerException is thrown	
		3- Applet2 is triggered	
2	Profile_Download is sent	33	
		1- Applet1 is triggered	
		2- NullPointerException is thrown	
		3- Applet2 is triggered	
3	UNRECOGNIZED_Envelope is sent		
		1- Applet1 is triggered	
		2- NullPointerException is thrown	
		3- Applet2 is triggered	
4	Event_Download_MT_Call is sent		
		1- Applet1 is triggered	
		2- NullPointerException is thrown	
		3- Applet2 is triggered	

ld	Description	API/Framework Expectation	APDU Expectation
5	Unformatted_SMS_PP_Env is sent		
		1- Applet1 is triggered : 2- NullPointerException is thrown	
		3- Applet2 is triggered	
6	Unformatted_SMS_PP_Upd is sent		
		1- Applet1 is triggered :	
		2- NullPointerException is thrown	
		3- Applet2 is triggered	
7	Unformatted_SMS_CB is sent		
		1- Applet1 is triggered	
		2- NullPointerException is thrown	
		3- Applet2 is triggered	

6.3.5.2.4 Test Coverage

CRR Number	Test Case Number	
CRRN1	1, 2, 3, 4, 5, 6, 7	

6.3.6 Framework Security Management

Security Parameters

The table that follows contains the security parameters that shall be used when the 3GPP TS 23.048 [8] security is required in the test cases developed in the current subclause.

Parameter	Value in hexadecimal	
KIC	11	
KID	11	
CNTR	00 00 00 00 01	
Key for ciphering	01 41 42 7F DA E8 91 A7	
Key for RC/CC/DS	01 23 45 67 89 AB CD EF	

If a parameter is not listed explicitly in the above table, the default values of subclause 4.7.3.1 apply.

6.3.6.1 Input Data

Test Area Reference: FWK_FWS_INDA

6.3.6.1.1 Conformance Requirements

6.3.6.1.1.1 Normal Execution

- CRRN1: If the SIM receives an envelope APDU containing an SMS_PP_DATADOWNLOAD BER TLV formatted according to 3GPP TS 23.048 [8], the SIM Toolkit Framework shall verify the security of the SMS TPDU.
- CRRN2: The toolkit applet will only be triggered if the TAR is known and the security verified.

- CRRN3: If the SIM receives an envelope APDU containing an SMS_CB_DATADOWNLOAD formatted according to 3GPP TS 23.048 [8], the SIM Toolkit Framework shall verify the security of the cell broadcast page.
- CRRN4: If the SIM receives an Update Record EFsms instruction formatted according to TS 23.048[8], the SIM Toolkit Framework shall verify the security of the SMS.
- CRRN5: The STF shall provide the input data deciphered.

6.3.6.1.1.2 Parameters error

No requirements.

6.3.6.1.1.3 Context Errors

No requirements.

6.3.6.1.2 Test Area Files

Test Script: FWK_FWS_INDA_1.scr

Test Applet: FWK_FWS_INDA_1.java

FWK_FWS_INDA_2.java

FWK_FWS_INDA_3.java

FWK_FWS_INDA_4.java

FWK_FWS_INDA_5.java

FWK_FWS_INDA_6.java

Load Script: FWK_FWS_INDA_1.ldr

Cleanup Script: FWK_FWS_INDA_1.clr

Parameter File: FWK_FWS_INDA_1.par

6.3.6.1.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation

412

ld	Description	API/Framework Expectation	APDU Expectation
1	•		1- The SIM answers to the Envelope with status words 9000
	paca rengen 15 150.		2- The SIM answers to the Envelope with status words 9000

ld	Description	API/Framework Expectation	APDU Expectation
2	Triggering two different applets with	-	-
	different security Applet2 is installed		
	1-Envelope(SMS-PP) single and formatted is sent to the SIM with this features: Ciphering; Cryptographic checksum; No proof of receipt; TAR of Applet 1 Data = 03	1- Applet 1 is triggered and the value integrity is checked	1- The SIM answers to the Envelope with status words 9000
	2- Short Message concatenated and formatted is sent to the SIM by an Envelope (SMS PP)with these features: Ciphering; Cryptographic checksum; No proof of receipt; TAR of Applet 1 Data length = 150	2- Applet 1 is triggered and the value integrity is checked	2- The SIM answers to the Envelope with status words 9000
	3-Envelope(SMS-PP) single and formatted is sent to the SIM with this features: No ciphering; No cryptographic checksum; No proof of receipt; TAR of Applet 2 Data = 05	3- Applet 2 is triggered and the value integrity is checked	3- The SIM answers to the Envelope with status words 9000
	4- Short Message concatenated and formatted is sent to the SIM by an Envelope (SMS PP)with these features:: No ciphering; No cryptographic checksum; No proof of receipt; TAR of Applet 2 Data length = 150.		4- The SIM answers to the Envelope with status words 9000
		4- Applet 2 is triggered and the value integrity is checked	
3	Envelope(SMS-PP) formatted with wrong cryptographic checksum		1- The SIM answers to the Envelope with status words 9000
	1-Envelope 03.48 single and formatted is sent to the SIM with this features: No ciphering; Wrong cryptographic checksum; No proof of receipt; TAR of Applet 1 Data = 07	1- No applet is triggered.	
	2- Short Message concatenated and formatted is sent to the SIM by an Envelope (SMS PP)with these features: No ciphering; Wrong cryptographic checksum; No proof of receipt; TAR of Applet 1 Data length = 150	2- No applet is triggered.	

ld	Description	API/Framework Expectation	APDU Expectation
4	Framework checks the Cryptographic checksum and deciphers the data		
	Applet3 is loaded and installed 1-Envelope(SMS-CB) formatted is sent to the SIM with this features: Ciphering; Cryptographic checksum; No proof of receipt; Data = 01	1- Applet3 is triggered and the value integrity is checked	1- The SIM answers to the Envelope with status words 9000
5	Triggering two different applets with different security on Envelope(SMS-CB) formatted Applet4 is installed 1-Envelope(SMS-CB) formatted is sent to the SIM with this features: Ciphering; Cryptographic checksum;	1- Applet3 is triggered and the value integrity is checked	1- The SIM answers to the Envelope with status words 9000
	No proof of receipt; TAR of Applet 3 Data = 02		
	2-Envelope(SMS-CB) formatted is sent to the SIM with this features: No ciphering; No cryptographic checksum; No proof of receipt; TAR of Applet 4 Data = 03	2- Applet4 is triggered and the value integrity is checked	2- The SIM answers to the Envelope with status words 9000
6	cryptographic checksum No ciphering; Wrong Cryptographic checksum; No proof of receipt;	No applet is triggered	1- The SIM answers to the Envelope with status words 9000
	TAR of Applet 3 Data = 04		

ld	Description	API/Framework Expectation	APDU Expectation
7	Framework checks the Cryptographic	·	•
	checksum and deciphers the data		
	Applet5 is installed		
	1- Short Message single and formatted is sent to the SIM by Update Record EFsms instruction with these features: Ciphering; Cryptographic checksum; No proof of receipt;	1- The applet5 is triggered and the value integrity is checked.	1- The SIM answers to the Update Record EFsms instruction with status words 9000
	TAR of Applet5; Data = 01		
	2- Short Message concatenated and formatted is sent to the SIM by Update Record EFsms instruction with these features: Ciphering; Cryptographic checksum; No proof of receipt;	2- The applet5 is triggered and the value integrity is checked	2- The SIM answers to the Update Record EFsms instruction with status words 9000
	TAR of Applet5;		
	Data length = 150.		
8	Triggering two different applets with different security		
	Applet6 is installed	1- Applet5 is triggered and the value integrity is checked.	1- The SIM answers to the Update Record EFsms instruction with status words 9000
	1- Short Message single and formatted is sent to the SIM by Update Record EFsms instruction with these features: Ciphering; Cryptographic checksum; No proof of receipt;		
	TAR of Applet5 Data = 03		2- The SIM answers to the
	2- Short Message concatenated and formatted is sent to the SIM by Update Record EFsms instruction with these features: Ciphering; Cryptographic checksum;	2- Applet5 is triggered and the value integrity is checked.	Update Record EFsms instruction with status words 9000
	No proof of receipt;		
	TAR of Applet5		
	Data length = 150. 3- Short Message single and formatted is sent to the SIM by Update Record EFsms	3- Applet6 is triggered and the	3- The SIM answers to the Update Record EFsms instruction with status words
	<pre>instruction with these features: No ciphering; No cryptographic checksum; No proof of receipt; TAR of Applet6; Data = 05</pre>	value integrity is checked.	9000
	4- Short Message concatenated and formatted is sent to the SIM by Update Record EFsms instruction with these features: No ciphering; No cryptographic checksum;	4- Applet6 is triggered and the value integrity is checked.	4- The SIM answers to the Update Record EFsms instruction with status words 9000
	No proof of receipt; TAR of Applet6; Data length = 150.		
9	Update Record EFsms instruction formatted		
	with wrong cryptographic checksum		
	1- Short Message single and formatted is sent to the SIM by Update Record EFsms		
	benc to the pru by obdate record Frams		Î

ld	Description	API/Framework Expectation	APDU Expectation
	<pre>instruction with these features:No ciphering; Wrong Cryptographic checksum; No proof of receipt; TAR of Applet5 Data = 07</pre>	1- No applet is triggered.	1- The SIM answers to the Update Record EFsms instruction with status words 9000
	2- Short Message concatenated and formatted is sent to the SIM by Update Record EFsms instruction with these features: No ciphering; Wrong Cryptographic checksum; No proof of receipt; TAR of Applet5 Data length = 150	2- No applet is triggered.	2- The SIM answers to the Update Record EFsms instruction with status words 9000

6.3.6.1.4 Test Coverage

CRR Number	Test Case Number
CRRN1	1, 2, 3
CRRN2	3,6,9
CRRN3	4, 5, 6
CRRN4	7,8,9
CRRN5	1,2,4,5,7,8

6.3.6.2 Output Data

Test Area Reference: FWK_FWS_OUDA

6.3.6.2.1 Conformance Requirements

6.3.6.2.1.1 Normal Execution

• CRRN1: The SIM Toolkit Framework shall secure and send the response packet.

6.3.6.2.1.2 Parameters error

No requirements.

6.3.6.2.1.3 Context Errors

No requirements.

6.3.6.2.2 Test Area Files

Test Script: FWK_FWS_OUDA_1.scr

Test Applet: FWK_FWS_OUDA_1.java

Load Script: FWK_FWS_OUDA_1.ldr

Cleanup Script: FWK_FWS_OUDA_1.clr

Parameter File: FWK_FWS_OUDA_1.par

6.3.6.2.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation

ld	Description	API/Framework Expectation	APDU Expectation
1	Envelope(SMS-PP) formatted Ciphering; Cryptographic checksum; proof of receipt response shall be sent using SMS-Deliver-Report; no security applied to proof of receipt Data in plain text = "APPLET1"	The applet is triggered and sends a "Display Text" proactive command with the data received in the Envelope.	The SIM answers to the Envelope with status words 9Fxx and a PoR is retrieved with a GetResponse command. The PoR has no application data. The SIM answers to the Get Response command with status words 91xx to issue a Display Text "APPLET1".
2	Envelope(SMS-PP) formatted Ciphering; Cryptographic checksum; proof of receipt response shall be sent using SMS-Deliver-Report; no security applied to proof of receipt Data in plain text = "APPLET1"	The applet posts application data. It does not call the ProactiveHandler.send() method	The SIM answers to the Envelope with status words 9Fxx and a PoR is retrieved with a GetResponse command. The PoR has the application data posted by the application. The SIM answers to the Get Response command with status words 9000.
3	<pre>Envelope(SMS-PP) formatted Ciphering; Cryptographic checksum; proof of receipt response shall be sent using SMS-Deliver-Report; no security applied to proof of receipt Data in plain text = "TEST"</pre>	The applet posts application data and calls the ProactiveHandler.send() method to send a "Display Text" proactive command with the data received in the Envelope.	The SIM answers to the Envelope with status words 9Fxx and a PoR is retrieved with a GetResponse command. The PoR has the application data posted by the application. The SIM answers to the Get Response command with status words 91xx to issue the Display Text "TEST".
4	Envelope(SMS-PP) formatted Ciphering; Cryptographic checksum; proof of receipt response shall be sent using SMS-Deliver-Report; proof of receipt shall be ciphered Data in plain text = "TEST"	The applet posts application data and calls the ProactiveHandler.send() method to send a "Display Text" proactive command with the data received in the Envelope.	The SIM answers to the Envelope with status words 9Fxx and a PoR is retrieved with a GetResponse command. The PoR has the application data posted by the application. The SIM answers to the Get Response command with status words 91xx to issue the Display Text "TEST".
5	Envelope(SMS-PP) formatted The Terminal Profile command shall be issued with the facility "'9EXX' response code for SIM data download error" enabled The Envelope(SMS-PP) formatted has to be issued with the following features: No ciphering; Wrong Cryptographic checksum; proof of receipt response shall be sent using SMS-Deliver-Report; no security applied to proof of receiptData in plain text = "TEST"	No applet is triggered	The SIM answers to the Envelope with status words 9Exx and a PoR is retrieved with a GetResponse command. The Response Status Code Octet shall be '01'.

6.3.6.2.4 Test Coverage

CRR Numb	er Test Case Number
CRRN1	1, 2, 3, 4, 5

6.3.7 Envelope Response Posting

6.3.7.1 EVENT_CALL_CONTROL_BY_SIM

Test Area Reference: FWK_ERP_ECCN

6.3.7.1.1 Conformance Requirements

6.3.7.1.1.1 Normal Execution

• CRRN1: The SIM Toolkit Framework can't reply busy when an Envelope(Call Control) is sent to the SIM.

6.3.7.1.1.2 Parameters error

No requirements.

6.3.7.1.1.3 Context Errors

No requirements.

6.3.7.1.2 Test Area Files

Test Script: FWK_ERP_ECCN_1.scr

Test Applet: FWK_ERP_ECCN_1.java

FWK_ERP_ECCN_2.java

FWK_ERP_ECCN_3.java

Load Script: FWK_ERP_ECCN_1.ldr

Cleanup Script: FWK_ERP_ECCN_1.clr

Parameter File: FWK_ERP_ECCN_1.par

6.3.7.1.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1	Applet 1 is registered on the EVENT_CALL_CONTROL_BY_SIM, Applet2 is registered and triggered on the EVENT_MENU_SELECTION.		
	1-Applet2 invokes the method send()and no fetch is performed 2-Envelope(Call Control) is sent to the	Applet2 is suspended	
	SIM 3-Applet1 calls the method EnvelopeResponseHandler.postASBERTLV() to change any incoming dialling number into +11 22 33 44.	Applet1 is triggered.	The SIM answer 9Fxx to the
			Envelope(Call Control) The dialling number is retrieved with a GetResponse command. The SIM answers to the Get Response command with status words 91xx.
	4-A Fetch command is sent to the SIM 5-A Terminal Response command is sent to		
	the SIM 6-Delete applet1 & applet2	Applet2's execution shall continue.	
	7-Install applet3		
2	Applet 3 is registered on both the events EVENT_CALL_CONTROL_BY_SIM and EVENT_MENU_SELECTION. 1-Envelope Menu Selection is sent to the SIM.	Applet3 is triggered on the EVENT_MENU_SELECTION	
	2-Applet3 invokes the method send()and no fetch is performed)	Applet3 is suspended on the send() method	
	3-Envelope(Call Control) is sent to the SIM 4-Applet3 calls the method	Applet3 is triggered on the EVENT_CALL_CONTROL_BY_SI	The CIM energy of the the
	EnvelopeResponseHandler.postASBERTLV() to change any incoming dialling number into +11 22 33 44.	M.	The SIM answer 9Fxx to the Envelope(Call Control) The dialling number is retrieved with a GetResponse command. The SIM answers to the Get Response command with status words 91xx.
	5-A Fetch command is sent to the SIM		
	6-A Terminal Response command is sent to the SIM		
		The Applet3's execution shall continue.	

6.3.7.1.4 Test Coverage

CRR Number	Test Case Number
CRRN1	1, 2

6.3.7.2 EVENT_MO_SHORT_MESSAGE_CONTROL_BY_SIM

Test Area Reference: FWK_ERP_EMCN

6.3.7.2.1 Conformance Requirements

6.3.7.2.1.1 Normal Execution

• CRRN1: The SIM Toolkit Framework can't reply busy when an Envelope(MO-Short Message Control) is sent to the SIM.

6.3.6.2.1.2 Parameters error

No requirements.

6.3.6.2.1.3 Context Errors

No requirements.

6.3.7.2.2 Test Area Files

Test Script: FWK_ERP_EMCN_1.scr

Test Applet: FWK_ERP_EMCN_1.java

FWK_ERP_EMCN_2.java

FWK_ERP_EMCN_3.java

Load Script: FWK_ERP_EMCN_1.ldr

Cleanup Script: FWK_ERP_EMCN_1.clr

Parameter File: FWK_ERP_EMCN_1.par

6.3.7.2.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1	Applet 1 is registered on the EVENT_MO_SHORT_MESSAGE_CONTROL_B Y_SIM; Applet2 is registered and triggered on the EVENT_MENU_SELECTION.		•
	1-Applet2 invokes the method send()and no fetch is performed) 2-Envelope(MO-SM control) is sent to the	Applet2 is suspended	
	3-Applet1 calls the method EnvelopeResponseHandler.postASBERTLV() to change any incoming TP_Destination_Address and any RP_Destination_Address of the Service Center into +11 22 33 44		The SIM answers 9Fxx to the Envelope(MO-Short
	4-A Fetch command is sent to the SIM		The TP_Destination_Address is retrieved with a GetResponse command. The SIM answers to the Get Response command with status words 91xx.
	5-A Terminal Response command is sent to the SIM	The Armhalla consolition shall	
	6-Delete applet1 & applet2	The Applet's execution shall continue.	
2	7-Install applet3 Applet 3 is registered on both the events EVENT_MO_SHORT_MESSAGE_CONTROL_B Y_SIM and EVENT_MENU_SELECTION.		
	1-Applet3 invokes the method send()and no fetch is performed)	Applet 3 is suspended on the send() method	
	2-Envelope(MO-SM control) is sent to the SIM 3-Applet3 calls the method EnvelopeResponseHandler.postASBERTLV() to	Applet3 is triggered on the EVENT_MO_SHORT_MESSAGE_CONTROL_BY_SIM.	
	change any incoming TP_Destination_Address and any RP_Destination_Address of the Service Center into +11 22 33 44.		The SIM answers 9Fxx to the Envelope(MO-Short Message Control)
			The TP_Destination_Address is retrieved with a GetResponse command.
			The SIM answers to the Get Response command with status words 91xx.
	4-A Fetch command is sent to the SIM		
	5-A Terminal Response command is sent to the SIM		
		The Applet3's execution shall continue.	

6.3.7.2.4 Test Coverage

CRR Number	Test Case Number
CRRN1	1, 2

6.3.7.3 EVENT_UNRECOGNIZED_ENVELOPE

Test Area Reference: FWK_ERP_EUEN

6.3.7.3.1 Conformance Requirements

6.3.7.3.1.1 Normal Execution

• CRRN1: The EnvelopeResponseHandler is available for the EVENT_UNRECOGNIZED_ENVELOPE.

6.3.7.3.1.2 Parameters error

No requirements.

6.3.7.3.1.3 Context Errors

No requirements.

6.3.7.3.2 Test Area Files

Test Script: FWK_ERP_EUEN_1.scr

Test Applet: FWK_ERP_EUEN_1.java

Load Script: FWK_ERP_EUEN_1.ldr

Cleanup Script: FWK_ERP_EUEN_1.clr

Parameter File: FWK_ERP_EUEN_1.par

6.3.7.3.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1	An applet triggered on the	The post() method returns no	The SIM answers to the
	EVENT_UNRECOGNIZED_ENVELOPE calls the	exception	Envelope with status words
	EnvelopeResponseHandler.post() method		9Fxx. The data retrieved
			with the GetResponse
			command are the ones
			posted by the applet.

6.3.7.3.4 Test Coverage

CRR Number	Test Case Number
CRRN1	1

6.3.7.4 EVENT_FORMATTED_SMS_PP_ENV

Test Area Reference: FWK_ERP_EFSE

6.3.7.4.1 Conformance Requirement

6.3.7.4.1.1 Normal Execution

- CRRN1: If PoR is required a SMS-DELIVER REPORT is sent by the SIM, when the post() or the postAsBERTLV() method is invoked and if bit 6 of the second octet of SPI is set to 0.
- CRRN2: If PoR is required a SMS-SUBMIT is sent by the SIM, when the post() or the postAsBERTLV() method is invoked and if bit 6 of the second octet of SPI is set to 1. In this case the statusType method parameter is meaningless. The SIM Toolkit Framework shall build and issue a Send Short Message proactive command as defined in TS 11.14 [4].

6.3.7.4.2 Test Suite Files

Test Script: FWK_ERP_EFSE_1.scr

Test Applet: FWK_ERP_EFSE_1.java

FWK_ERP_EFSE _2.java

Load Script: FWK_ERP_EFSE _1.ldr

Cleanup Script: FWK_ERP_EFSE _1.clr

Parameter File: FWK_ERP_EFSE _1.par

424

6.3.7.4.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1	SMS DELIVER REPORT		•
	1- A formatted sms pp envelope with SMS Deliver Report required is sent to the SIM with bit 6 of SPI2 set to 0.	1- Applet1 is triggered	
	2- EnvelopeResponseHandler.getTheHandler() method is called by Applet1	2- No exception is thrown.	
	3- Applet1 builds the answer and calls the post() method with StatusType=SW1_RP_ACK	Applet1 finalizes	3- ME receives 9FXX and checks the response
	4- A formatted sms pp envelope with SMS Deliver Report required is sent to the SIM with bit 6 of SPI2 set to 0.		
	5- EnvelopeResponseHandler.getTheHandler() method is called by Applet1 6- Applet1 builds the answer and calls the postAsBERTLV() method with StatusType=SW1_RP_ACK	4- Applet1 is triggered	5- ME receives 9FXX and checks the response
		E. No expension is thrown	
		5- No exception is thrown.	
		Applet1 finalizes	
2	SMS-SUBMIT		
	1- A formatted sms pp envelope with SMS Submit required is sent to the SIM with bit 6 of SPI2 set to 1.	1- Applet1 is triggered	
	2- EnvelopeResponseHandler.getTheHandler() method is called by Applet1	2- No exception is thrown.	
	3-Applet1 builds the answer and calls the post() method with StatusType=SW1_RP_ACK	Applet1 finalizes	3- ME receives a Send Short Message proactive command.
	4- A formatted sms pp envelope with SMS Submit required is sent to the SIM with bit 6 of SPI2 set to 1.	4- Applet1 is triggered	
	5- EnvelopeResponseHandler.getTheHandler() method is called by Applet1	5- No exception is thrown	

ld	Description	API/Framework Expectation	APDU Expectation
	6-Applet1 builds the answer and calls the post() method with StatusType=SW1_RP_ERROR	Applet1 finalizes	6- ME receives a Send Short Message proactive command.
	7- A formatted sms pp envelope with SMS Submit required is sent to the SIM with bit 6 of SPI2 set to 1. 8- EnvelopeResponseHandler.getTheHandler() method is called by Applet1	7- Applet1 is triggered	
	9Applet1 builds the answer and calls the postAsBERTLV() method with StatusType=SW1_RP_ACK	8- No exception is thrown.	9- ME receives a Send Short Message proactive command.
		Applet1 finalizes	
	10- A formatted sms pp envelope with SMS Submit required is sent to the SIM with bit 6 of SPI2 set to 1.		
	11- EnvelopeResponseHandler.getTheHandler() method is called by Applet1	10- Applet1 is triggered	
	12-Applet1 builds the answer and calls the postAsBERTLV () method with StatusType=SW1_RP_ERROR	11- No exception is thrown.	12- ME receives a Send Short Message proactive command.

6.3.7.4.4 Test Coverage

CRR Number	Test Case Number	
CRRN1	1	
CRRN2	2	

6.3.8 Toolkit Installation

6.3.8.1 Timers Allocation

Test Area Reference: FWK_TIN_TMAL

6.3.8.1.1 Conformance Requirements

6.3.8.1.1.1 Normal execution

• CRRN1: One toolkit applet can register to several timers, but a timer can only be allocated to one toolkit applet.

6.3.8.1.1.2 Parameters error

No requirements.

6.3.8.1.1.3 Context errors

- CRRC1: Allocated timers shall not exceed the maximum number of timers allowed for this applet instance defined during installation.
- CRRC2: The total number of timers allocated for all the applets shall not exceed 8. If the maximum number of timers required is greater than '08' (maximum numbers of timers specified in TS 11.14 [4], the card shall return the Status Word '6A80', incorrect parameters in data field, to the Install(Install) command.

6.3.8.1.2 Test suite files

Test Script: FWK_TIN_TMAL_1.scr

Test Applet: FWK_TIN_TMAL_1.java

FWK_TIN_TMAL_2.java

FWK_TIN_TMAL_3.java

Load Script: FWK_TIN_TMAL_1.ldr

Cleanup Script: FWK_TIN_TMAL_1.clr

Parameter File: FWK_TIN_TMAL_1.par

6.3.8.1.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1	More than 8 timers at the instantiation of applet1: check that applet1 is not installed. Install for install of applet1 with maximum 9 timers allocated, requesting a POR to be sent via SMS-DELIVER-REPORT.		
			The SIM answers to the Envelope with status words 9Fxx
			A GET RESPONSE is sent and the additional data in the PoR is checked. It must be 01 6A 80.
	Reset the card		
2	Good installation of applet2 Install for install of applet2 (maximum 4 timers allocated).		The SIM answers to the Envelope with status words 90 00
3	Allocate 4 timers Applet2	No exception shall be thrown.	
4	Allocate one more timer Applet2	Shall throw a ToolkitException with reason NO_TIMER_AVAILABLE	
5	Good installation of applet3		

ld	Description	API/Framework Expectation	APDU Expectation
	Install for install of applet3 (maximum 8 timers allocated).		The SIM answers to the Envelope with status words 90 00
6	Allocate 4 timers Applet3	No exception shall be thrown.	
7	Allocate one more timer Applet3	Shall throw a ToolkitException with reason NO_TIMER_AVAILABLE	
8	Check that each timerId (allocated by applet2 and applet3) is between 1 and 8 and is different from each other		

6.3.8.1.4 Test Coverage

CRR number	Test case number	
N1	2, 3, 8	
C1 1, 7		
C2	4, 5, 6	

6.3.8.2 Item Identifier

Test Area Reference: FWK_TIN_ITID

6.3.8.2.1 Conformance Requirements

6.3.8.2.1.1 Normal execution

- CRRN1: If the requested item identifier in the range [1-127] is not already allocated, then this item identifier shall be allocated to the current applet.
- CRRN2: If the requested item identifier is '00', the card shall take the first free value in the range [128,255].

6.3.8.2.1.2 Parameters error

• CRRP1: If the requested item identifier is in the range [128,255], then the card shall reject the install command.

6.3.8.2.1.3 Context errors

• CRRC1: If the requested item identifier in the range [1-127] is already allocated, then the card shall reject the install command.

6.3.8.2.2 Test suite files

Test Script: FWK_TIN_ITID_1.scr

Test Applet: FWK_TIN_ITID_1.java

FWK_TIN_ITID_2.java

FWK_TIN_ITID_3.java

Load Script: FWK_TIN_ITID_1.ldr

Cleanup Script: FWK_TIN_ITID_1.clr

Parameter File: FWK_TIN_ITID_1.par

6.3.8.2.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1	Bad installation of applet1 Install for install of applet1.The following parameters item Id equal to 128		applet1 is not found, status word 6X XX
2	Good installation of applet1		
	<pre>Install for install of applet1. item Id = 1 for the first menu and 127 for the second one</pre>		The SIM answers to the Envelope with status words 91xx to send back to the ME the 2 new menus.
	A Terminal Profile is sent to the card with only PROFILE_DOWNLOAD, SMS_PP_DOWNLOAD, MENU_SELECTION, SET_UP_MENU and COMMAND_RESULT facilities.		The menus are (position/itemId/text) 01/01/menu11 02/127/menu12
3	Bad installation of applet2		
	Item identifier already allocated		
	<pre>Install for install of applet2. item Id = 127 applet2 is selected</pre>		applet2 is not found, status word 6X XX
4	Good installation of applet2		
	<pre>Install for install of applet2. item Id = 0</pre>		The SIM answers to the Envelope with status words 91xx to send back to the ME the 3 menus.
			The menus are 01/01/menu11 02/127/menu12 03/128/menu21
5	Good installation of applet3		
	<pre>Install for install of applet3. item Id = 0</pre>		The SIM answers to the Envelope with status words 91xx to send back to the ME the 4 menus.
			The menus are 01/01/menu11 02/127/menu12 03/128/menu21 04/129/menu31

ld	Description	API/Framework Expectation	APDU Expectation
6	Good delete and installation of applet2		
	Delete instance of applet2		The SIM answers to the Terminal Profile with status
	Perform a RESET and a Terminal Profile with the facilities of PROFILE_DOWNLOAD,		words 91xx to send back to the ME the 3 menus.
	SMS-PP_DATA_DOWNLOAD, MENU_SELECTION, COMMAND_RESULT and SET_UP_MENU		The menus are 01/01/menu11 02/127/menu12 03/129/menu31
	Install for install of applet2.		
	item Id = 0		The SIM answers to the Envelope with status words 91xx to send back to the ME the 4 menus.
			The menus are 01/01/menu11 02/127/menu12 03/128/menu21 04/129/menu31

6.3.8.2.4 Test Coverage

CRR number	Test case number
N1	2
N2	4, 5, 6
P1 1	
C1	3

6.3.8.3 Item Position

Test Area Reference: FWK_TIN_ITPO

6.3.8.3.1 Conformance Requirements

6.3.8.3.1.1 Normal execution

- CRRN1: The position of the new menu entries is an absolute position among the existing ones.
- CRRN2: If the position identifier is 00h, the menu shall have the last position.

6.3.8.3.1.2 Parameters error

No requirements.

6.3.8.3.1.3 Context errors

No requirements.

6.3.8.3.2 Test suite files

Test Script: FWK_TIN_ITPO_1.scr

Test Applet: FWK_TIN_ITPO_1.java

FWK_TIN_ITPO_2.java

FWK_TIN_ITPO_3.java

Load Script: FWK_TIN_ITPO_1.ldr

Cleanup Script: FWK_TIN_ITPO_1.clr

Parameter File: FWK_TIN_ITPO_1.par

6.3.8.3.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1	Installation of applet1	•	•
	Perform Install for install of applet1.Position/ItemId 01/01 02/02		
	A Terminal Profile is sent to the card		The menus are (position/itemId/text) 01/01/menu11 02/02/menu12
2	Installation of applet2 Perform Install for install of applet2. Position/ItemId 03/03		The SIM answers to the Envelope with status words 91xx to send back to the ME the 4 menus.
	04/04		The menus are (position/itemId/text) 01/01/menu11 02/02/menu12 03/03/menu21 04/04/menu22
3	Installation of applet3 Perform Install for install of applet3. Position/ItemId 00/05		The SIM answers to the Envelope with status words 91xx to send back to the ME the 5 menus. The menus are (position/itemId/text) 01/01/menu11 02/02/menu12 03/03/menu21 04/04/menu22
			04/04/menu22 05/05/menu31

6.3.8.3.4 Test Coverage

NOTE: As Item Position management is not fully specified in the 3GPP TS 43.019 [7] or 3GPP TS 23.048 [8] all possible tests cannot be performed.

CRR number	Test case number	
N1	1, 2	
N2	3	

6.3.8.4 Maximum Text Length for a menu entry

Test Area Reference: FWK_TIN_MLME

6.3.8.4.1 Conformance Requirements

6.3.8.4.1.1 Normal execution

• CRRN1: The maximum length of item text string is defined at the installation of the toolkit applet.

6.3.8.4.1.2 Parameters errors

- CRRP1: If initMenuEntry length parameter is greater than the allocated space (Maximum Text Length for a menu entry), then a ToolkitException ALLOWED_LENGTH_EXCEEDED is thrown.
- CRRP2: If changeMenuEntry length parameter is greater than the allocated space (Maximum Text Length for a menu entry), then a ToolkitException ALLOWED_LENGTH_EXCEEDED is thrown.

6.3.8.4.1.3 Context errors

No requirements.

6.3.8.4.2 Test suite files

Test Script: FWK_TIN_MLME_1.scr

Test Applet: FWK_TIN_MLME_1.java

Load Script: FWK_TIN_MLME_1.ldr

Cleanup Script: FWK_TIN_MLME_1.clr

Parameter File: FWK_TIN_MLME_1.par

6.3.8.4.3 Test Procedure

ld	Description	API / Framework Expectation	APDU Expectation
1	Installation of applet with 2 menus not exceeding the maximum text length		
	Install one applet with 2 menu entries allowed and max. text length equal to 10. initMenuEntry defined at the install (install) command MenuEntry = "MenuEntry1", "MenuEntry2" Offset = 0 Length = 10 NextAction = '00'		
	<pre>HelpSupported = false IconQualifier = '00' IconIdentifier = 0</pre>		
2	<pre>initMenuEntry with a too large length initMenuEntry with length equal to 11 MenuEntry = " MenuEntry03" Offset = 0 Length = 11 NextAction = '00' HelpSupported = false IconQualifier = '00' IconIdentifier = 0</pre>	ToolkitException ALLOWED_LENGTH_EXCEEDED is thrown	
3	<pre>initMenuEntry with a right length initMenuEntry with length parameter equal to 10 MenuEntry = " MenuEntry3" Offset = 0 Length = 10 NextAction = '00' HelpSupported = false IconQualifier = '00' IconIdentifier = 0</pre>		a SET UP MENU (2 items) is issued with TLV item length equal to 11 (Identifier + Text string of item)

ld	Description	API / Framework Expectation	APDU Expectation
4	changeMenuEntry with a right length Applet1 is triggered by a EVENT_MENU_SELECTION. changeMenuEntry of menu 1, with length parameter equal to 10 Id = '01' MenuEntry = "MenuEntry4" Offset = 0 Length = menuEntry.length NextAction = 0 HelpSupported = false IconQualifier = 0 IconIdentifier = 0		a SET UP MENU (2 items) is issued with TLV item length equal to 11 (Identifier + Text string of item)
_	Return from processToolkit		0
5	changeMenuEntry with a too large length Applet1 is triggered by a EVENT_MENU_SELECTION. ChangeMenuEntry of menu 1, with length parameter equal to 11 Id = '02' MenuEntry = "MenuEntry05" Offset = 0 Length = menuEntry.length NextAction = 0 HelpSupported = false IconQualifier = 0 IconIdentifier = 0 Return from processToolkit	ToolkitException ALLOWED_LENGTH_EXCEEDED is thrown	Shall not receive a SET UP MENU different from the previous one

6.3.8.4.4 Test Coverage

CRR number	Test case number	
CRRN1	1, 3, 4	
CRRP1	2	
CRRP2	5	

6.3.8.5 Maximum number of menu entries

Test Area Reference: FWK_TIN_NBME

6.3.8.5.1 Conformance Requirements

6.3.8.5.1.1 Normal execution

• CRRN1: The maximum number of menu entries is defined at the installation of the toolkit applet and can be the maximum number of successful invocations of the method initMenuEntry.

6.3.8.5.1.2 Parameters errors

• CRRP1: If the menu entry cannot be initialised (e.g. no more item data in applet loading parameter), a ToolkitException with the REGISTRY_ERROR reason code is thrown.

6.3.8.5.1.3 Context errors

No requirements.

6.3.8.5.2 Test suite files

Test Script: FWK_TIN_NBME_1.scr

Test Applet: FWK_TIN_NBME_1.java

FWK_TIN_NBME_2.java

Load Script: FWK_TIN_NBME_1.ldr

Cleanup Script: FWK_TIN_NBME_1.clr

Parameter File: FWK_TIN_NBME_1.par

6.3.8.5.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1 1	<pre>Installation of applet with 3 menus Install (install) applet with max. number of menu entry is '3', defined at the install (install) command. initMenuEntry for each menu entry allowed (3 times) MenuEntry = "menul", "menu2", "menu3" Offset = 0 Length = 5 NextAction = '00'</pre>	API/Framework Expectation No Exception is thrown	APDU Expectation
	<pre>HelpSupported = false IconQualifier = '00' IconIdentifier = 0</pre>		
2	<pre>init of a 4th menu initMenuEntry one more time MenuEntry = "menu4" Offset = 0 Length = 5 NextAction = '00' HelpSupported = false IconQualifier = '00' IconIdentifier = 0</pre>	ToolkitException REGISTRY_ERROR is thrown	SET UP MENU (3 items) is issued with TLV item length equal to 6 (Identifier + Text string of item)
3	Installation of 2 nd applet with 0 menu Install (install) another applet, with max. number of menu entry is '0', defined at the install (install) command. initMenuEntry once MenuEntry = "menul" Offset = 0 Length = 5 NextAction = '00' HelpSupported = false IconQualifier = '00' IconIdentifier = 0	ToolkitException REGISTRY_ERROR is thrown	Shall not receive a SET UP MENU different from the previous one

6.3.8.5.4 Test Coverage

CRR number	R number Test case number	
CRRN1	1	
CRRP1	2, 3	

6.3.8.6 Access Domain

Test Area Reference: FWK_TIN_ACDO

6.3.8.6.1 Conformance Requirements

6.3.8.6.1.1 Normal execution

• CRRN1: The Access Domain parameter indicates the mechanism used to control the applet instance access to the GSM file System ('00' means full access to the GSM File System, 'FF' means no access to the GSM File System).

6.3.8.6.1.2 Parameters errors

- CRRP1: If the Access Domain Parameter requested is not supported, the card shall return the Status Word '6A80', incorrect parameters in data field, to the Install(Install) command.
- CRRP2: If an applet with Access Domain Parameter 'FF' (i.e. No Access to the GSM File System) tries to access a GSM file (e.g. invoke the updateBinary(..) method) the framework shall throw a SIMViewException with a AC_NOT_FULFILLED reason.

6.3.8.6.1.3 Context errors

No requirements.

6.3.8.6.2 Test suite files

Test Script: FWK_TIN_ACDO_1.scr

Test Applet: FWK_TIN_ACDO_1.java

FWK_TIN_ACDO_2.java

FWK_TIN_ACDO_3.java

Load Script: FWK_TIN_ACDO_1.ldr

Cleanup Script: FWK_TIN_ACDO_1.clr

Parameter File: FWK_TIN_ACDO_1.par

6.3.8.6.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
0	Install (install) applet1 with:		
	- Length of Access Domain field value is		
	'1'		
	- Access Domain Parameter value is '00'		
	(full access to the GSM File System)		
	Install (install) applet2 with:		
	- Length of Access Domain field value is		
	'1'		
	- Access Domain Parameter value is 'FF'		
	(No access to the GSM File System)		
	Install (install) applet3 with:		
	- Length of Access Domain field value is		
	'1'		
	- Access Domain Parameter value is '00'		
	(full access to the GSM File System)		

ld	Description	API/Framework Expectation	APDU Expectation
1	readBinary/readRecord method with full	1 to 4- no exception is thrown	<u>-</u>
	Access Domain Parameter	'	
		5- SIMViewException	
	1- Select EF-TARU file whose Read access	AC_NOT_FULFILLED is thrown	
	condition is ALWAYS	//O_ITOT_I OEI IEEED IO UIIOWII	
	Perform the readBinary method:		
	fileOffset = 0		
	resp = abRead[]		
	respOffset = 0		
	respLength = 3		
	2- Select EF-SMS file whose Read access		
	condition is CHV1		
	Perform the readRecord method:		
	recNumber = 1		
	mode = REC_ACC_MODE_ABSOLUTE_CURRENT		
	recOffset = 0		
	resp = abRead[]		
	respOffset = 0		
	respLength = 3		
	3- Select EF-TRAC file whose Read access		
	condition is CHV2		
	Perform the readBinary method:		
	fileOffset = 0		
	resp = abRead[]		
	respOffset = 0		
	respLength = 3		
	4- Select EF-SUME file Read access		
	condition is ADM0		
	Perform the readBinary method:		
	fileOffset = 0		
	resp = abRead[]		
	respOffset = 0		
	respLength = 3		
	5- Select EF-TNR file whose Read access		
	condition is NEVER		
	Perform the readBinary method:		
	fileOffset = 0		
	resp = abRead[]		
	respOffset = 0		
	respLength = 3		

ld	Description	API/Framework Expectation	APDU Expectation
2	updateBinary/updateRecord method with full	1 to 4- no exception is thrown	•
	Access Domain Parameter		
	For each case, send an Envelope that	E CIMVious voontion	
	triggers the applet with the	5- SIMViewException AC_NOT_FULFILLED is thrown	
	EVENT_UNFORMATTED_SMS_PP_ENV event.	AO_NOT_FOR TELED IS UNOWIT	
	1- Select EF-TNR file whose Update access		
	condition is ALWAYS		
	Perform the updateBinary method: fileOffset = 0		
	resp = abUpdate[FFFFFF]		
	respOffset = 0		
	respLength = 3		
	2- Select EF-SMS file whose Update access		
	condition is CHV1		
	Perform the updateRecord method: recNumber = 1		
	mode = REC_ACC_MODE_ABSOLUTE_CURRENT		
	recOffset = Oresp = abUpdate[]		
	respOffset = 0 respLength = 3		
	respicingen = 5		
	3- Select EF-FDN file whose Update access		
	condition is CHV2 Perform the updateBinary method:		
	recNumber = 1		
	<pre>mode = REC_ACC_MODE_ABSOLUTE_CURRENT recOffset = 0</pre>		
	resp = abUpdate[]		
	respOffset = 0		
	respLength = 3		
	4- Select EF-SUME file Update access		
	condition is ADM0		
	Perform the updateBinary method: fileOffset = 0		
	resp = abUpdate[]		
	respOffset = 0		
	respLength = 3		
	5- Select EF-TNU file whose Update access		
	condition is NEVER		
	Perform the updateBinary method: fileOffset = 0		
	resp = abUpdate[]		
	respOffset = 0		
	respLength = 3		
3	invalidate method with full Access Domain	1 to 4- no exception is thrown	
	Parameter		
		5- SIMViewException	
	1- Select EF-TNR file whose Invalidate	AC_NOT_FULFILLED is thrown	
	access condition is ALWAYS Perform the invalidate method		
	Terrorm ene invaridade mediod		
	2- Select EF-TIAC file whose Invalidate		
	access condition is CHV1 Perform the invalidate method		
	3- Select EF-ADN file whose Invalidate access condition is CHV2		
	Perform the invalidate method		
	4- Select EF-SUME file Invalidate access		
	condition is ADM0 Perform the invalidate method		
	5- Select EF-CNIV file whose Invalidate		
	access condition is NEVER Perform the invalidate method		
L			
L			I .

ld	Description	API/Framework Expectation	APDU Expectation
4	rehabilitate method with full Access Domain	1 to 4- no exception is thrown	
	Parameter	·	
	1- Select EF-TNR file whose Rehabilitate access condition is ALWAYS	5- SIMViewException AC_NOT_FULFILLED is thrown	
	Perform the rehabilitate method 2- Select EF-IMSI file whose Rehabilitate		
	access condition is CHV1 Perform the rehabilitate method		
	3- Select EF-ADN file whose Rehabilitate access condition is CHV2 Perform the rehabilitate method		
	4- Select EF-SUME file Rehabilitate access condition is ADMO Perform the rehabilitate method		
	5- Select EF-CNRI file whose Rehabilitate access condition is NEVER Perform the rehabilitate method		
5	increase method with full Access Domain Parameter	1 to 4- no exception is thrown	
	<pre>1- Select EF-CNU file whose Increase access condition is ALWAYS Perform the increase method: incr = abIncreaseValue[] incrOffset = 0 resp = abRead[] respOffset = 0</pre>	5- SIMViewException AC_NOT_FULFILLED is thrown	
	2- Select EF-ACM file whose Increase access condition is CHV1 Perform the increase method: incr = abIncreaseValue[] incrOffset = 0 resp = abRead[] respOffset = 0		
	3- Select EF-CIAC file whose Increase access condition is CHV2 Perform the increase method: incr = abIncreaseValue[] incrOffset = 0 resp = abRead[] respOffset = 0		
	4- Select EF-CIAA file Increase access condition is ADMO Perform the increase method: incr = abIncreaseValue[] incrOffset = 0 resp = abRead[] respOffset = 0		
	5- Select EF-CNR file whose Increase access condition is NEVER Perform the increase method		

ld	Description	API/Framework Expectation	APDU Expectation
6	readBinary method with no Access Domain	SIMViewException	
	Parameter	AC_NOT_FULFILLED is thrown	
	Sond an Envolone that triggory the applict		
	Send an Envelope that triggers the applet with the EVENT_UNFORMATTED_SMS_PP_ENV		
	event.		
	Select EF-TARU file whose Read access condition is ALWAYS		
	Perform the readBinary method:		
	fileOffset = 0		
	resp = abRead[]		
	respOffset = 0 respLength = 3		
	t		
7	updateRecord method with no Access Domain		
	Parameter	AC_NOT_FULFILLED is thrown	
	Send an Envelope that triggers the applet		
	with the EVENT_UNFORMATTED_SMS_PP_ENV		
	event.		
	Select EF-SMS file whose Update access		
	condition is CHV1		
	Perform the updateRecord method:		
	<pre>fileOffset = 0 resp = abUpdate[]</pre>		
	respOffset = 0		
	respLength = 3		
	involtate math - death Access Book	OIM () (in the first of the fir	
8	invalidate method with no Access Domain Parameter	SIMViewException AC_NOT_FULFILLED is thrown	
	raiametei	AC_NOT_I OLI ILLED IS UIIOWII	
	Send an Envelope that triggers the applet		
	with the EVENT_UNFORMATTED_SMS_PP_ENV event.		
	event.		
	Select EF-ADN file whose Invalidate access		
	condition is CHV2 Perform the invalidate method		
	reflorm the invalidate method		
9	rehabilitate method with no Access Domain	SIMViewException	
	Parameter	AC_NOT_FULFILLED is thrown	
	Send an Envelope that triggers the applet		
	with the EVENT_UNFORMATTED_SMS_PP_ENV		
	event.		
	Select EF-SUME file Rehabilitate access		
	condition is ADMO		
	Perform the rehabilitate method		
10	increase method with no Access Domain	SIMViewEveention	
10	Parameter	SIMViewException AC_NOT_FULFILLED is thrown	
	- 2.2		
	Send an Envelope that triggers the applet		
	with the EVENT_UNFORMATTED_SMS_PP_ENV event.		
	evene.		
	Select EF-CNR file whose Increase access		
	condition is NEVER Perform the increase method		
	relional the increase alethod		
		Applet2 finalizes	
		Applet3 restore EF-SUME	

6.3.8.6.4 Test Coverage

NOTE: As Item Position management is not fully specified in the 3GPP TS 43.019 [7] or 3GPP TS 23.048 [8] all possible tests cannot be performed.

CRR number	Test case number
CRRN1	1, 2, 3, 4, 5
CRRP1	Not tested
CRRP2	6, 7, 8, 9, 10

6.3.8.7 Priority Level

Test Area Reference: FWK_TIN_PRLV

6.3.8.7.1 Conformance Requirements

6.3.8.7.1.1 Normal execution

- CRRN1: The priority specifies the order of activation of an applet compared to the other applet registered to the same event ('01': Highest priority level, 'FF': Lowest priority level).
- CRRN2: If two or more applets are registered to the same event and have the same priority level, the applets are activated according to their installation date (i.e. the most recent applet is activated first).

6.3.8.7.1.2 Parameters errors

No requirements.

6.3.8.7.1.3 Context errors

No requirements.

6.3.8.7.2 Test suite files

Test Script: FWK_TIN_PRLV_x.scr, x from 1 to 12

Test Applet: FWK_TIN_PRLV_x.java, x from 1 to 12, 8A, 8B, 9A, 9B, 10A, 10B

Load Script: FWK_TIN_PRLV_x.ldr, x from 1 to 12

Cleanup Script: FWK_TIN_PRLV_x.clr, x from 1 to 12

Parameter File: FWK_TIN_PRLV_x.par, x from 1 to 12, 8A, 8B, 9A, 9B, 10A, 10B

6.3.8.7.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
0	All applets are registered on an		
	EVENT_UNFORMATTED_SMS_PP_ENV event		
1	Trigger 2 applets with 2 different maximum		
	Priority Levels		
	Install (install) applet1 with priority level '2' and applet2 with priority level '1', from package fwk_tin_prlv_1.		
	Send an Envelope that triggers the 2 applets with the EVENT_UNFORMATTED_SMS_PP_ENV event.	A static variable is used to validate triggering order: applet2 is triggered before applet1	
	Delete applets instances and packages		

ld	Description	API/Framework Expectation	APDU Expectation
2	Trigger 2 applets with 2 different maximum		
	Priority Levels		
	To the 11 (in the 11) and 1st 1 with and with		
	<pre>Install (install) applet1 with priority level '1' and applet2 with priority level</pre>		
	'2', from package fwk_tin_prlv_2.		
	Send an Envelope that triggers the 2 applets with the		
	EVENT_UNFORMATTED_SMS_PP_ENV event.		
		A static variable is used to validate	
	Delete applets instances and packages	triggering order: applet1 is	
	11 3	triggered before applet2.	
3	Trigger 2 applets with 2 different Priority Levels		
	LOVOIS		
	Install (install) applet1 with priority		
	level '80' and applet2 with priority level '7F', from package fwk_tin_prlv_3.		
	Fr , IIOM package IWK_tIM_priv_3.		
	Send an Envelope that triggers the 2		
	applets with the		
	EVENT_UNFORMATTED_SMS_PP_ENV event.		
	Delete applets instances and packages	A static variable is used to validate	
		triggering order: applet2 is	
		triggered before applet1	
4	Trigger 2 applets with 2 different Priority Levels		
	Leveis		
	Install (install) applet1 with priority		
	level '7F' and applet2 with priority level		
	'80', from package fwk_tin_prlv_4.		
	Send an Envelope that triggers the 2		
	applets with the		
	EVENT_UNFORMATTED_SMS_PP_ENV event.		
		A static variable is used to validate	
		triggering order: applet2 is	
	Delete applets instances and packages	triggered before applet1	
5	Trigger 3 applets with the same Priority Level		
	<pre>Install (install) applet 1, 2, 3 in this order with same priority level from</pre>		
	package fwk_tin_prlv_5.		
	Send an Envelope that triggers the 3 applets with the		
	EVENT_UNFORMATTED_SMS_PP_ENV event.		
	Delete applets instances and packages.	A static variable is used to validate	
		triggering order: applet3 is triggered before applet2, and	
		applet2 is triggered before applet1.	
6	Trigger 2 applets from 2 classes, with 2	E-F-10-10 mggorod poloto applotti	
	different Priority Level		
	Tuntall (in tall) and 11 5		
	Install (install) applet1 from class A with priority level '2'		
	Install (install) applet2 from class B		
	with priority level '1'		
	Send an Envelope that triggers the 2		
	applets with the		
	EVENT_UNFORMATTED_SMS_PP_ENV event.		
	Delete applets instances and packages	A static variable is used to validate	
		triggering order: applet2 is	
		triggered before applet1	

ld	Description	API/Framework Expectation	APDU Expectation
7	Trigger 2 applets from 2 classes, with the same Priority Level		
	,		
	Install (install) applet1 from class A with priority level '1'		
	Install (install) applet2 from class B with priority level '1'		
	Send an Envelope that triggers the 2		
	applets with the EVENT_UNFORMATTED_SMS_PP_ENV event.		
	Delete applets instances and packages	A static variable is used to validate	
		triggering order: applet2 is	
		triggered before applet1	
8	Trigger 2 applets from 2 packages, with 2 different Priority Level		
	Install package fwk_tin_prlv_8.		
	Install (install) applet1 from package fwk_tin_prlv_8A with priority level '2'		
	Install (install) applet2 from package fwk_tin_prlv_8B with priority level '1'		
	Send an Envelope that triggers the 2 applets with the		
	EVENT_UNFORMATTED_SMS_PP_ENV event.		
	Delete applets instances ad packages	A static variable is used to validate triggering order: applet2 is	
	Trimman O annulate from O marks was with the	triggered before applet1	
9	Trigger 2 applets from 2 packages, with the same Priority Level		
	Install package fwk_tin_prlv_9.		
	Install (install) applets 1 from package		
	fwk_tin_prlv_9A and applet2 from package fwk_tin_prlv_9B in this order, with same		
	priority level		
	Send an Envelope that triggers the 2 applets with the EVENT_UNFORMATTED_SMS_PP_ENV event.		
		A static variable is used to validate	
	Delete applets instances and packages	triggering order: applet2 is triggered before applet1	

ld	Description	API/Framework Expectation	APDU Expectation
10	Trigger 4 applets from 2 packages		
	1-Install packages fwk_tin_prlv_10, fwk_tin_prlv_10B and fwk_tin_prlv_10B. Install (install) 2 applets 1 then 2 from package fwk_tin_prlv_10A, with respectively priority levels 1 and 2.		
	Send an Envelope that triggers the 2 applets with the EVENT_UNFORMATTED_SMS_PP_ENV event.		
	2- Install (install) 2 applets 3 then 4 from package fwk_tin_prlv_10B, with respectively priority levels 1 and 2.		
	Send an Envelope that triggers the 4 applets.	1- A static variable is used to validate triggering order: applet1 is triggered before applet2	
	Delete applets instances and packages		
		2- Applet3 is triggered before applets 1, 4, then 2.	
11	Trigger 4 applets with the same Priority Level		
	then delete them one after another and trigger them each time		
	1- Install (install) applet1, 2, 3, 4 in this order with same priority level from package fwk_tin_prlv_11.		
	Send an Enveloppe that triggers the 4 applets with the EVENT_UNFORMATTED_SMS_PP_ENV event.		
	Delete applet instance 4	1- A static variable is used to validate triggering order: applets are triggered in order 4, 3, 2, 1.	
	2- Send an Enveloppe that triggers the 3 applets with the EVENT_UNFORMATTED_SMS_PP_ENV event.		
	Delete applet instance 3		
	3- Send an Enveloppe that triggers the 2 applets with the EVENT_UNFORMATTED_SMS_PP_ENV event.	2- Applets are triggered in order 3, 2, 1.	
	Delete remaining applet instances and packages		
		3- Applets are triggered in order 2, 1.	

ld	Description	API/Framework Expectation	APDU Expectation
12	Trigger 5 applets with different Priority Levels, alternating install and delete		
	1- Install (install) applets 1, 2, 3, 4 in this order with respective priority levels 1, 2, 1, 2		
	Send an Enveloppe that triggers the 4 applets with the EVENT_UNFORMATTED_SMS_PP_ENV event.		
		1- A static variable is used to validate triggering order: applets are triggered in order 3, 1, 4, 2	
	2- Delete applet instance 1 and install (install) applet5 with priority level 2	are inggered in order 3, 1, 4, 2	
	Send an Enveloppe that triggers the 4 applets with the EVENT_UNFORMATTED_SMS_PP_ENV event.		
	3- Re-install (install) applet1 with priority level 1	2- Applets are triggered in order 3, 5, 4, 2	
	Send an Enveloppe that triggers the 5 applets with the EVENT_UNFORMATTED_SMS_PP_ENV event.		
		3- Applets are triggered in order 1, 3, 5, 4, 2	

6.3.8.7.4 Test Coverage

CRR number	Test case number
CRRN1	1, 2, 3, 4, 6, 8, 10, 12
CRRN2	5, 7, 9, 11

6.3.8.8 Channel Allocation

Test Area Reference: FWK_TIN_CHAL

6.3.8.8.1 Conformance Requirements

6.3.8.8.1.1 Normal execution

• CRRN1: One toolkit applet can register to several channels, but a channel can only be allocated to one toolkit applet.

6.3.8.8.1.2 Context errors

- CRRC1: Allocated channels shall not exceed the maximum number of channels allowed for this applet instance.
- CRRC2: The total number of channels allocated for all the applets shall not exceed 7. If the maximum number of channels required is greater than '07' (maximum numbers of channels specified in TS 11.14 [4]), the card shall return the Status Word '6A80', incorrect parameters in data field, to the Install(Install) command.

6.3.8.8.2 Test suite files

Test Script: FWK_TIN_CHAL_1.scr

Test Applet: FWK_TIN_CHAL_1.java

FWK_TIN_CHAL_2.java

FWK_TIN_CHAL_3.java

Load Script: FWK_TIN_CHAL_1.ldr

Cleanup Script: FWK_TIN_CHAL_1.clr

Parameter File: FWK_TIN_CHAL_1.par

6.3.8.8.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1	More than 7 channels at the instantiation of applet1: check that applet1 is not installed		
	1-Install for install of applet1 with maximum 8 channels allocated. A POR is asked to be sent via SMS-DELIVER-REPORT.		1- The SIM answers to the Envelope with status words 9Fxx. A GET RESPONSE is sent and the additional data in the PoR is checked. It must be 01 6A 80.
	Reset the card		
2	Good installation of applet2		
	Install for install of applet2 (maximum 4 channels allocated).		The SIM answers to the Envelope with status words 90 00
3	Open 4 channels Applet2	No exception shall be thrown.	OPEN CHANNEL proactive command are fetched.
	Applet2 builds a proactive command OPEN CHANNEL 4 times, calling init() and send() methods.		Successful TERMINAL RESPONSE of OPEN CHANNEL are sent to the SIM with Channel Id = 01 to 04
4	Open one more channel	Shall throw a ToolkitException with	
	Applet2	reason	
	Applet2 builds a proactive command OPEN CHANNEL once again, calling init() and send() methods.	COMMAND_NOT_ALLOWED	
5	Good installation of applet3		
	Install for install of applet3 (maximum 7 channels allocated).		The SIM answers to the Envelope with status words 90 00
6	Open 3 channels Applet3	No exception shall be thrown.	OPEN CHANNEL proactive command is fetched.
	Applet3 builds a proactive command OPEN CHANNEL 3 times, calling init() and send() methods.		Successful TERMINAL RESPONSE of OPEN CHANNEL are sent to the SIM with Channel Id from 05 to 07
7	Open one more channel Applet3	No exception shall be thrown.	OPEN CHANNEL proactive command is fetched. Unsuccessful Terminal
	Applet3 builds a proactive command OPEN CHANNEL once again, calling init() and send() methods.		Response is sent to the SIM with 'No Channel Available' as Additional Information on Result.

6.3.8.8.4 Test Coverage

CRR number	Test case number
N1	2,3
C1	1, 7
C2	4,5,6

6.3.8.9 Minimum Security Level

Test Area Reference: FWK_TIN_MSL

6.3.8.9.1 Conformance Requirements

6.3.8.9.1.1 Normal execution

- CRRN1: The Receiving Entity shall check the Minimum Security Level during processing the security of the Command Packet.
- CRRN2: The Receiving Entity shall reject the message if the MSL check fails.
- CRRN3: If the MSL check fails, a Response Packet with the 'Insufficient Security Level' Response Status Code shall be sent if required.
- CRRN4: If the length of the Minimum Security Level field is greater than zero, the Minimum Security Level is used to specify the minimum level of security to be applied to Secured Packets. The first byte shall be the MSL Parameter, other bytes shall be the MSL Data.
- CRRN5: If the length of the Minimum Security Level field is zero, no minimum security level check shall be performed by the receiving entity.
- CRRN6: If no Minimum Security Level field is present (no MSL length, no MSL parameter and no MSL data), no minimum security level check shall be performed by the receiving entity.
- CRRN7: If the Maximum number of channels field is included in the command data then the Length of Minimum Security Level field shall also be included.
- CRRN8: If an optional parameter is included, then all the previous parameters shall be included also

6.3.8.9.2 Test suite files

Test Script: FWK_TIN_MSL_1.scr

Test Applet: FWK_TIN_MSL_1.java

Load Script: FWK_TIN_MSL_1.ldr

Cleanup Script: FWK_TIN_MSL_1.clr

Parameter File: FWK_TIN_MSL_1.par

6.3.8.9.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1	Installation with MSL length of 0		
	1- Install (install) applet with a MSL length = 0 2- Send formatted SMS PP env with no RC/CC/DS, no Ciphering and counter mode 0	2- Applet is triggered	1- 9000
	<pre>(not checked) 3- Send a formatted SMS PP env with CC, ciphering and counter mode 1 (counter available and no checking) 4- Delete the applet instance</pre>	3- Applet is triggered	
2	Installation without MSL field		
	1- Install (install) applet without MSL field (no MSL length, no MSL parameter and		1- 9000
	no data) 2- Send formatted SMS PP env with no RC/CC/DS, no Ciphering and counter mode 0	2- Applet is triggered	
	(not checked) 3- Send a formatted SMS PP env with CC,	3- Applet is triggered	
	ciphering and counter mode 1 counter available and no checking) 4- Delete the applet instance		

6.3.8.9.4 Test Coverage

CRR number	Test case number	
CRRN1	Not applicable	
CRRN2	Not applicable	
CRRN3	Not applicable	
CRRN4	Not applicable	
CRRN5	1	
CRRN6	2	
CRRN7	Not testable	
CRRN8	Not testable	

6.3.9 File System Context

6.3.9.1 Initial Context

Test Area Reference: FWK_FSC_INIT

6.3.9.1.1 Conformance Requirements

6.3.9.1.1.1 Normal Execution

• CRRN1: At the invocation of the processToolkit method of a toolkit applet, the current file is the MF.

6.3.9.1.1.2 Parameters errors

No requirements.

6.3.9.1.1.3 Context errors

No requirements.

6.3.9.1.2 Test Suite Files

Test Script: FWK_FSC_INIT_1.scr

Test Applet: FWK_FSC_INIT_1.java

FWK_FSC_INIT_2.java

Load Script: FWK_FSC_INIT_1.ldr

Cleanup Script: FWK_FSC_INIT_1.clr

FWK_FSC_INIT_2.clr

Parameter File: FWK_FSC_INIT_1.par

6.3.9.1.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
0	SIM Initialization	Responses ignored.	
1	<pre>MF is the selected DF in processToolkit() An ENVELOPE APDU containing a formatted SMS PP for Applet 1 is issued to the SIM byte[] fci = new byte[10] fciOffset = 0 fciLength = 7 status()</pre>	No exception shall be thrown. Shall return 7. fci shall contain the following part of the FCI structure: < XX XX XX XX 3F 00 01 >	
2	No EF is selected rehabilitate ()	SIMView exception shall be thrown with reason NO EF SELECTED	
3	MF is selected even when an applet triggered before selected any other file Applets 1 and 2 register to EVENT_DOWNLOAD_USER_ACTIVITY. Applet 1 has higher priority than Applet 2. An ENVELOPE "EVENT - USER ACTIVITY" is sent to the SIM 1 - Applet 1: - is triggered by event_event_download_user_activity - selects DF_GSM and EF_IMSI 2 - Applet 2: - is triggered by event_event_download_user_activity fciOffset = 0 fciLength = 7 status() 3 - rehabilitate ()	No exception shall be thrown. No exception shall be thrown. Shall return 7.	

6.3.9.1.4 Test Coverage

CRR Number	Test Case Number
CRRN1	1, 2, 3

6.3.9.2 Context Preservation (current file)

Test Area Reference: FWK_FSC_CUFI

6.3.9.2.1 Conformance Requirements

6.3.9.2.1.1 Normal execution

- CRRN1: When calling the method select (), the current files (file context) of any other applets shall not be changed (see 3GPP TS 43.019 [7] subclause 5.2).
- CRRN2: The select() methods select a file without changing the current file of any other applet or of the subscriber session.

• CRRN3: After invocation of ProactiveHandler.send() method: the current file context of the toolkit applet is unchanged (see 3GPP TS 43.019 [7] - subclause 5.2.).

6.3.9.2.1.2 Parameters errors

No requirements.

6.3.9.2.1.3 Context errors

No requirements.

6.3.9.2.2 Test Suite Files

Test Script: FWK_FSC_CUFI_1.scr

Test Applet: FWK_FSC_CUFI_1.java

FWK_FSC_CUFI_2.java

Load Script: FWK_FSC_CUFI_1.ldr

Cleanup Script: FWK_FSC_CUFI_1.clr

 $FWK_FSC_CUFI_2.clr$

Parameter File: FWK_FSC_CUFI_1.par

6.3.9.2.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
0	SIM Initialization	Responses ignored.	
1	No change to file context by another applet Applet1 registers to EVENT_FORMATTED_SMS_PP_ENV. Applet2 registers to EVENT_CALL_CONTROL_BY_SIM	3 - No exception shall be thrown.	A GET INKEY proactive command is fetched from the SIM
	<pre>1 - Applet 1: - is triggered by a formatted SMS - selects DF_SIMTEST and EF_TARU - fileOffset = 0; dataLength = 2; dataOffset = 0; - buffer = {0xCA, 0xFE } - updateBinary (): first 2 bytes of EF_TARU are written as 'CA FE' issues a proactive command "Get Inkey".</pre>		
	2 - An ENVELOPE APDU containing a CALL CONTROL BY SIM is issued to the SIM Applet 2: - is triggered by a CALL CONTROL BY SIM		
	- selects DF_TELECOM and EF_ADN. 3 - The terminal response for Get Inkey reactivates Applet 1: - fileOffset = 0; respLength = 2; respOffset = 0; - readBinary () info buffer2		

ld	Description	API/Framework Expectation	APDU Expectation
2	No change to file context by subscriber session 1 - Applet 1 - issues a proactive command "Get Inkey". 2 - Subscriber session selects DF_TELECOM and EF_ADN. 3 - The terminal response for Get Inkey reactivates Applet 1: - fileOffset = 0; respLength = 2; respOffset = 0; readBinary () info buffer2	1 - No exception shall be thrown. 3 - No exception shall be thrown. The value of buffer2 is { 0xCA, 0xFE }	1 - A GET INKEY proactive command is fetched from the SIM
3	No change by applet of subscriber session context 1 - Applet 1: - selects DF_SIMTEST and EF_TNU - issues a proactive command "Get Inkey". 2 - subscriber session reads record 1 of current file (shall be EF_ADN) 3 - The terminal response for Get Inkey reactivates Applet 1, which terminates execution	No exception shall be thrown. No exception shall be thrown.	1 - A GET INKEY proactive command is fetched from the SIM 2 - READ RECORD absolute number 1 shall read "FF FF

6.3.9.2.4 Test Coverage

CRR Number	Test Case Number	
CRRN1	1	
CRRN2	1, 2, 3	
CRRN3	1,2	

6.3.9.3 Context Preservation (current record pointer)

Test Area Reference: FWK_FSC_CURE

6.3.9.3.1 Conformance Requirements

6.3.9.3.1.1 Normal execution

- CRRN1: When the seek method is called by one applet, the record pointer of any other applet is not changed.
- CRRN2: *updateRecord*: the current record pointer of other applets / subscriber shall not be changed in case of linear fixed EF
- CRRN3: *updateRecord*: the record pointer of a cyclic EF shall be changed for all other applets / subscriber to the record number 1.
- CRRN4: *readRecord*: read data bytes of the linear fixed or cyclic EF currently selected by the applet without changing the current record pointer of any other applet / subscriber.
- CRRN5: *increase*: the last updated record of the cyclic EF currently selected becomes record number 1 for every other applet and subscriber session.

6.3.9.3.1.2 Parameters errors

No requirements.

6.3.9.3.1.3 Context errors

No requirements.

6.3.9.3.2 Test Suite Files

Test Script: FWK_FSC_CURE_1.scr

Test Applet: FWK_FSC_CURE_1.java

FWK_FSC_CURE_2.java

Load Script: FWK_FSC_CURE_1.ldr

Cleanup Script: FWK_FSC_CURE_1.clr

 $FWK_FSC_CURE_2.clr$

Parameter File: FWK_FSC_CURE_1.par

6.3.9.3.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
0	SIM Initialization	Responses ignored.	
1	Seek without affecting another record pointer Applet1 registers to EVENT_FORMATTED_SMS_PP_ENV Applet 2 registers to EVENT_CALL_CONTROL_BY_SIM	1 - No exception shall be thrown.2 - No exception shall be thrown.3 - No exception shall be thrown.	1 - A GET INKEY proactive command is fetched from the SIM
	1 - Applet 1: - is triggered by a formatted SMS event - selects DF_SIMTEST and EF_LARU - reads record 2 using NEXT so that the current record pointer is set to record 2 - issues a proactive command, e.g. Get Inkey.		
	2 - An ENVELOPE APDU containing a CALL CONTROL BY SIM is issued to the SIM		
	Applet 2: - is triggered by a CALL CONTROL event - selects DF_SIMTEST and EF_LARU - performs a seek of pattern {0x55} from beginning forward, which finds record 1 returns from processToolkit		
	3 - The terminal response for Get Inkey reactivates Applet 1: - call readRecord() using CURRENT - the record read should still be record 2 of EF_LARU, containing {0xAA, 0xAA, 0xAA, 0xAA}		

ld	Description	API/Framework Expectation	APDU Expectation
2	updateRecord in linear fixed EF without	1 - No exception shall be thrown.	1 - A GET INKEY proactive
	affecting current pointer of others	2 - No exception shall be thrown.	command is fetched from
	<pre>1 - Applet 1: - is triggered by a formatted SMS event - selects DF_SIMTEST and EF_LARU - reads record 2 using NEXT so that the current record pointer is set to record 2 - issues a proactive command, e.g. Get Inkey.</pre>	3 - No exception shall be thrown.	the SIM
	2 - An ENVELOPE APDU containing a CALL CONTROL BY SIM is issued to the SIM		
	Applet 2: - is triggered by a CALL CONTROL BY SIM event - selects DF_SIMTEST and EF_LARU - updates record 1, by using mode "NEXT" returns from processToolkit		
	3 - The terminal response for Get Inkey reactivates Applet 1: - call readRecord() using CURRENT - the record read should still be record 2 of EF_LARU, containing {0xAA, 0xAA, 0xAA, 0xAA}		
3	readRecord in linear fixed EF without affecting	1 - No exception shall be thrown.	1 - A GET INKEY proactive
	current pointer of others 1 - Applet 1: - is triggered by a formatted SMS event - selects DF_SIMTEST and EF_LARU - reads record 2 using NEXT so that the current record pointer is set to record 2 - issues a proactive command, e.g. Get Inkey.	2 - No exception shall be thrown. 3 - No exception shall be thrown.	command is fetched from the SIM
	2 - An ENVELOPE APDU containing a CALL CONTROL BY SIM is issued to the SIM		
	Applet 2: - is triggered by a CALL CONTROL BY SIM event - selects DF_SIMTEST and EF_LARU - reads record 1, by using mode "NEXT" returns from processToolkit		
	3 - The terminal response for Get Inkey reactivates Applet 1: - call readRecord() using CURRENT - the record read should still be record 2 of EF_LARU, containing {0xAA, 0xAA, 0xAA, 0xAA}		

6.3.9.3.4 Test Coverage

С	RR Number	Test Case Number
	CRRN1	1
	CRRN2	2
	CRRN3	not tested (see note)
CRRN4		3
CRRN5		not tested (see note)
NOTE: These requirements have not been tested because of an inconsistent behaviour in		
3GPP TS 43.019 [7], which is foreseen to be corrected in future releases.		

6.3.10 Other parts transferred to framework from API

6.3.10.1 A handler is a temporary JCRE Entry Point object

Test Area Reference: FWK_API_HEPO

6.3.10.1.1 Conformance Requirement:

6.3.10.1.1.1 Normal execution

- CRRN1: The EnvelopeHandler is a Temporary JCRE Entry Point Object (see Javacard 2.1 Runtime Environment (JCRE) Specification [12]).
- CRRN2: The EnvelopeResponseHandler is a Temporary JCRE Entry Point Object (see Javacard 2.1 Runtime Environment (JCRE) Specification [12]).
- CRRN3: The ProactiveHandler is a Temporary JCRE Entry Point Object (see Javacard 2.1 Runtime Environment (JCRE) Specification [12]).
- CRRN4: The ProactiveResponseHandler is a Temporary JCRE Entry Point Object (see Javacard 2.1 Runtime Environment (JCRE) Specification [12]).

6.3.10.1.1.2 Parameters errors

No requirements.

6.3.10.1.1.3 Context errors

No requirements.

6.3.10.1.2 Test suite files

Test Script: FWK_API_HEPO_1.scr

Test Applet: FWK_API_HEPO_1.java

Load Script: FWK_API_HEPO_1.ldr

Cleanup Script: FWK_API_HEPO_1.clr

Parameter File: FWK_API_HEPO_1.par

6.3.10.1.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1	EnvelopeHandler.getTheHandler and store it in	SecurityException is thrown	
	a static field of the toolkit applet		
2	EnvelopeHandler.getTheHandler and store it in	SecurityException is thrown	
	a field of the toolkit applet		
3	EnvelopeResponseHandler.getTheHandler and	SecurityException is thrown	
	store it in a static field of the toolkit applet		
4	EnvelopeResponseHandler.getTheHandler and	SecurityException is thrown	
	store it in a field of the toolkit applet		
5	ProactiveHandler.getTheHandler and store it in	SecurityException is thrown	
	a static field of the toolkit applet		
6	ProactiveHandler.getTheHandler and store it in	SecurityException is thrown	
	a field of the toolkit applet		
7	Build and send a DISPLAY TEXT command to		
	be able to get the reference of the		Proactive command fetched
	ProactiveReponseHandler		and terminal response is
			issued
	ProactiveResponseHandler.getTheHandler and	SecurityException is thrown	
	store it in a static field of the toolkit applet		

ld	Description	API/Framework Expectation	APDU Expectation
8	ProactiveResponseHandler.getTheHandler and	SecurityException is thrown	
	store it in a field of the toolkit applet		

6.3.10.1.4 Test Coverage

CRR number Test case numbe	
N1	1, 2
N2	3, 4
N3	5, 6
N4	7, 8

6.3.10.2 Transaction

Test Area Reference: FWK_API_TRAN

6.3.10.2.1 Conformance Requirement:

6.3.10.2.1.1 Normal execution

• CRRN1: A pending toolkit applet transaction at the ProactiveHandler.send() method invocation is aborted.

6.3.10.2.1.2 Parameters errors

No requirements.

6.3.10.2.1.3 Context errors

No requirements.

6.3.10.2.2 Test suite files

Test Script: FWK_API_TRAN_1.scr

Test Applet: FWK_API_TRAN_1.java

Load Script: FWK_API_TRAN_1.ldr

Cleanup Script: FWK_API_TRAN_1.clr

Parameter File: FWK_API_TRAN_1.par

6.3.10.2.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1	Verify that transaction is aborted when a proactive command is sent		
	Initialise a byte field with 0x05		
	Build a display text proactive command.		
	beginTransaction()		
	Update the byte with 0x02		
	send the proactive command		
			Proactive command fetched
			and terminal response is issued
	Verify that the byte value is 0x05		issueu
	JCSystem.getTransactionDepth()	Shall return 0	

6.3.10.2.4 Test Coverage

CRR number	Test case number
N1	1

6.3.10.3 Timer Id between Applets

Test Area Reference: FWK_API_TMID

6.3.10.3.1 Conformance Requirement:

6.3.10.3.1.1 Normal execution

No requirements.

6.3.10.3.1.2 Parameters errors

No requirements.

6.3.10.3.1.3 Context errors

• CRRC1: The method ToolkitRegistry.releaseTimer() shall throw a ToolkitException with INVALID_TIMER_ID reason if the timer is valid but isn't allocated to this applet.

6.3.10.3.2 Test suite files

Test Script: FWK_API_TMID_1.scr

Test Applet: FWK_API_TMID_1.java

Load Script: FWK_API_TMID_1.ldr

Cleanup Script: FWK_API_TMID_1.clr

Parameter File: FWK_API_TMID_1.par

6.3.10.3.3 Test Procedure

Id	Description	API/Framework Expectation	APDU Expectation
1	During installation :		
	First instance allocate a timer and store the		
	returned value in a static field.		
	Second instance allocate a timer.		
	Trig second instance and try to releaseTimer()	releaseTimer() shall throw a	
	with the static field value.	ToolkitException with	
		INVALID_TIMER_ID reason	

6.3.10.3.4 Test Coverage

CRR number	Test case number
N1	1

6.3.11 Concatenated SMS

6.3.11.1 Concatenation processing

6.3.11.1.1 Conformance Requirements:

6.3.11.1.1.1 Normal execution

• CRRN1: The SIM Toolkit Framework shall link single Short Messages together to re-assemble the original message before any further processing.

455

- CRRN2: The concatenation control headers used to re-assemble the short messages in the correct order shall not be present in the SMS TPDU.
- CRRN3: The TP-elements of the SMS TPDU and the Address (TS-Service-Centre-Address) shall correspond to the ones in the last received Short Message (independently of the Sequence number of Information-Element-Data).
- CRRN4: The original Short Message shall be placed in one SMS TPDU TLV (with TP-UDL field coded on one octet) included in the EnvelopeHandler.
- CRRN5: The SIM Toolkit Framework shall be able to process messages with the following properties:
- The Information Element Identifier is equal to the 8-bit reference number

FWK_CSM_PROC_1.par

It contains uncompressed 8 bit data or uncompressed UCS2 data.

6.3.11.2 **Test Suite Files**

Test Script: FWK_CSM_PROC_1.scr Test Applet: FWK_CSM_PROC_1.java Load Script: FWK_CSM_PROC_1.ldr Cleanup Script: FWK_CSM_PROC_1.clr Parameter File:

6.3.11.3 Test Procedure 6.3.11.2

ld	Description	API/Framework Expectation	APDU Expectation
	Applet registration to EVENT_FORMATTED_SMS_PP_ENV and triggering		
	Applet is registered to EVENT_FORMATTED_SMS_PP_ENV and EVENT_UNFORMATTED_SMS_PP_ENV		
	A concatenated formatted SMS_PP short message is sent to the SIM (composed of two segments).		
1	The second segment of a concatenated short message is sent to the SIM.	Applet is not triggered.	
2	The first segment of the concatenated short message is sent to the SIM	Applet is triggered.	
3	Call the EnvelopeHanlder.getTheHandler()	No exception is thrown.	
4	Call the EnvelopeHandler.findTLV()to select the Dev Id, the adress and the TPDU TLV and the EnvelopeHandler.compareValue() to check each content.	Check that the message has been re-assembled in the correct order. Check that TP-UDL field is coded one octet. Check that the concatenation control header is not present in	

		the message. Check the	
5	A new concatenated formatted short message is	integrity of the message. Applet is triggered.	
5	sent to the SIM composed of two segments. The Address field of the first segment is different from the address field in the second segment.	Appier is triggered.	
6	Call the EnvelopeHandler.getTheHandler()	No exception is thrown.	
7	Call the EnvelopeHandler.findTLV()to select the the address TLV and the EnvelopeHandler.compareValue() to check its content. A new concatenated formatted short message is	Check that the address field of the message is equal to the address field of the second segment.	
0	sent to the SIM composed of two segments. Some TP_elements of the TP_DU of the first segment are different from the TP elements in the second segment.	Applet is triggered.	
9	Call the EnvelopeHandler.getTheHandler()	No exception is thrown.	
10	Call the EnvelopeHandler.findTLV()to select the the TP DU TLV and the EnvelopeHandler.compareValue() to check its TP elements.	Check that the TP elements of the message are equal to the ones of the second segment.	
11	Send a concatenated formatted short message (composed of 2 segment) with uncompressed 8 bits data.	Applet is triggered.	
	Applet registration to EVENT_UNFORMATTED_SMS_PP_ENV and triggering Same test as 1 but with an unformatted SMS_PP		
	envelope. A concatenated unformatted SMS_PP short message		
	is sent to the SIM (composed of two segments).		
12	The second segment of a concatenated short message is sent to the SIM.	Applet is not triggered.	
13	The first segment of the concatenated short message is sent to the ${\tt SIM}$	Applet is triggered.	
14	Call the EnvelopeHanlder.getTheHandler()	No exception is thrown.	
15	Call the EnvelopeHandler.findTLV()to select the Dev Id, the adress and the TPDU TLV and the EnvelopeHandler.compareValue() to check each content.	Check that the message has been re-assembled in the correct order. Check that TP-UDL field is coded one octet. Check that the concatenation control header is not present in the message. Check the integrity of the message.	
16	A new concatenated formatted short message is sent to the SIM composed of two segments. The Address field of the first segment is different from the address field in the second segment.	Applet is triggered.	
17	Call the EnvelopeHandler.getTheHandler()	No exception is thrown.	
18	Call the EnvelopeHandler.findTLV()to select the the address TLV and the EnvelopeHandler.compareValue() to check its content.	Check that the address field of the message is equal to the address field of the second segment.	
19	A new concatenated unformatted short message is sent to the SIM composed of two segments. Some TP_elements of the TP_DU of the first segment are different from the TP_elements in the second segment.	Applet is triggered.	
20	Call the EnvelopeHandler.getTheHandler()	No exception is thrown.	
21	Call the EnvelopeHandler.findTLV()to select the the TP DU TLV and the EnvelopeHandler.compareValue() to check its TP elements.	Check that the TP elements of the message are equal to the ones of the second segment.	
22	Send a concatenated unformatted short message (composed of 2 segments) with uncompressed UCS2	Applet is triggered.	

data.

6.3.11.4 Test Coverage

CRR number	Test case number
N1	1,2, 3, 5, 6, 8, 9, 12, 13,
	14, 16, 17, 19, 20
N2	4,15
N3	7,10, 18, 21
N4	4,15
N5	11,22

Annex A (normative): Class and Methods AID numbering and acronyms

A.1 Sim.access

Class Name	Acronyms	Numbering on 5 bits
SIMView	SVW	00001
SIMSystem	SSY	00010
SIMViewException	SVE	00011

A.1.1 SIMView methods

Method Name	Acronyms	Numbering on 6 bits
static final Constants		000001
short increase(byte[] incr, short incrOffset, byte[]	INCR_BS_BS	000010
resp, short respOffset)		
void invalidate()	INVL	000011
<pre>void readBinary(short fileOffset, byte[] resp, short</pre>	REDBS_BSS	000100
respOffset, short respLength)		
short readRecord(short recNumber, byte mode, short	REDRSBS_BSS	000101
recOffset, byte[] resp, short respOffset, short		
respLength)		
<pre>void rehabilitate()</pre>	REHA	000110
<pre>short seek(byte mode, byte[] patt, short pattOffset,</pre>	SEEKB_BSS	000111
short pattLength)		
<pre>void select(short fid)</pre>	SLCTS	001000
short select(short fid, byte[] fci, short fciOffset,	SLCTS BSS	001001
short fciLength)	_	
short status(byte[] fci, short fciOffset, short	STAT_BSS	001010
fciLength)	_	
short updateBinary(short fileOffset, byte[] data,	UPDBS_BSS	001011
short dataOffset, short dataLength)		
<pre>void updateRecord(short recNumber, byte mode, short</pre>	UPDRSBS_BSS	001100
recOffset, byte[] data, short dataOffset, short		
dataLength)		

A.1.2 SIMSystem methods

Method Name	Acronyms	Numbering on 6 bits
static SIMView getTheSIMView()	GETS	000001

A.1.3 SIMViewException methods

Method Name	Acronyms	Numbering on 6 bits
static void throwIt(short reason)	THITS	000001
SIMViewException(short reason)	COORS	000010
Constants	CONS	000011

A.2 Sim.toolkit

Class Name	Acronyms	Numbering on 5 bits
ToolkitConstants	TKC	00001
ToolkitInterface	TKI	00010

Class Name	Acronyms	Numbering on 5 bits
EditHandler	EDH	00011
EnvelopeHandler	ENH	00100
EnvelopeResponseHandler	ERH	00101
MEProfile	MEP	00110
ProactiveHandler	PAH	00111
ProactiveResponseHandler	PRH	01000
ToolkitRegistry	TKR	01001
ViewHandler	VWH	01010
ToolkitException	TKE	01011

A.2.1 ToolkitConstants

	Method Name	Acronyms	Numbering on 6 bits
I	Constants	CONS	000001

A.2.2 ToolkitInterface methods

Method Name	Acronyms	Numbering on 6 bits
<pre>void processToolkit (byte event)</pre>	PRTKB	000001

A.2.3 EditHandler methods

The numbering of the EditHandler methods it will be done in the classes inherit it: EnvelopeResponseHandler, ProactiveHandler, because the methods provided by this class as it is declared 'abstract'.

A.2.4 EnvelopeHandler methods

Method Name	Acronyms	Numbering on 6 bits
Byte getEnvelopeTag()	GENT	000001
Byte getItemIdentifier()	GIID	000010
Short getSecuredDataLength()	GSDL	000011
Short getSecuredDataOffset()	GSDO	000100
EnvelopeHandler getTheHandler()	GTHD	000101
Short getTPUDLOffset()	GTPO	000110
Short getCapacity()	GCAP	010010
Short getUserDataLength()	GUDL	010011
Byte getChannelIdentifier()	GCID	010100
Inherited Method Name: ViewHandler		
Byte	CPRVS_BSS	000111
<pre>compareValue(short valueOffset,byte[] compareBuffer,s</pre>		
hort compareOffset, short compareLength)		
Short	COPY_BSS	001000
<pre>copy(byte[] dstBuffer,short dstOffset,short dstLength</pre>		
Short.	CPYVS_BSS	001001
copyValue(short valueOffset,	CF1V3_B33	001001
byte[] dstBuffer, short dstOffset, short dstLength)		
Byte	FACRB BS	001010
<pre>findAndCompareValue(byte tag,byte[] compareBuffer,sho</pre>	-	
rt compareOffset)		
Byte findAndCompareValue(byte tag,byte occurrence,	FACRBBS BSS	001011
short valueOffset,byte[] compareBuffer,short compareO	FACRDDS_DSS	001011
ffset, short compareLength)		
Short	FACYBBS BSS	001100
FindAndCopyValue(byte tag,byte occurrence,short value	· · · · · · · · · · · · · · · ·	
Offset, byte[] dstBuffer, short dstOffset,		

Method Name	Acronyms	Numbering on 6 bits
short dstLength)		
Short	FACYB_BS	001101
<pre>findAndCopyValue(byte tag,byte[] dstBuffer,short dst0</pre>	_	
ffset)		
Byte	FINDBB	001110
FindTLV(byte tag,byte occurrence)		
Short	GLEN	001111
GetLength()		
Byte	GVBYS	010000
GetValueByte(short valueOffset)		
Short	GVLE	010001
GetValueLength()		

A.2.5 EnvelopeResponseHandler methods

Method Name	Acronym	Numbering on 6 bits
EnvelopeResponseHandler getTheHandler()	GTHD	000001
Void post(byte statusType)	POSTB	000010
Void postAsBERTLV(byte statusType, byte tag)	POSTBB	000011
Short getCapacity()	GCAP	010101
3	00/11	010101
Inherited Method Name: EditHandler		
Void appendArray(byte[] buffer, short offset, short	APDA_BSS	000100
<pre>length, short dstLength) Void appendTLV(byte tag, byte value)</pre>	APTLBB	000101
Void appendTLV(byte tag, byte[] value, short		000101
valueOffset, short valueLength)	APTLB_BSS	000110
Void appendTLV(byte tag, byte value1, byte value2)	APTLBBB	000111
Void appendTLV(byte tag, byte value1, byte[] value2,	APTLBB BSS	001000
short value2Offset, short value2Length)		
Void clear()	CLER	001001
Inherited Method Name: ViewHandler		
Byte	CPRVS_BSS	001010
<pre>compareValue(short valueOffset,byte[] compareBuffer,s</pre>		
hort compareOffset, short compareLength)		
Short	COPY_BSS	001011
<pre>Copy(byte[] dstBuffer,short dstOffset,short dstLength)</pre>		
Short	CPYVS_BSS	001100
CopyValue(short valueOffset,	0 0_200	
byte[] dstBuffer,short dstOffset,short dstLength)		
Byte	FACRB_BS	001101
findAndCompareValue(byte tag,byte[] compareBuffer,sho	FACRD_DS	001101
rt compareOffset)		
Byte findAndCompareValue(byte tag,byte occurence,	FACRBBS_BSS	001110
short valueOffset,byte[] compareBuffer,short compareO		
ffset,short compareLength)		
Short	FACYBBS_BSS	001111
<pre>findAndCopyValue(byte tag,byte occurence,short value0 ffset, byte[] dstBuffer, short dstOffset,</pre>		
short dstLength)		
Short.	FACYB_BS	010000
<pre>findAndCopyValue(byte tag,byte[] dstBuffer,short dst0</pre>	TACTB_BS	010000
ffset)		
Byte	FINDBB	010001
findTLV(byte tag,byte occurrence)		
Short	GLEN	010010
GetLength()		
Byte	GVBYS	010011
getValueByte(short valueOffset) Short	0)// 5	040400
getValueLength()	GVLE	010100
Accountered in ()		

A.2.6 MEProfile methods

Method Name	Acronym	Numbering on 6 bits
static boolean check(byte index)	CHECB	000001
static boolean check(byte[] mask, short offset, short	CHECBSS	000010
length)		
static boolean check(short index)	CHECS	000011
static short copy(short startOffset,	COPYS BSS	000100
<pre>byte[] dstBuffer, short dstOffset, short dstLength)</pre>		
static short getValue(short indexMSB, short indexLSB)	GVALSS	000101

A.2.7 ProactiveHandler methods

Method Name	Acronyms	Numbering on 6 bits
ProactiveHandler getTheHandler()	GTHD	000001
Void init(byte type, byte qualifier, byte dstDevice)	INITBBB	000010
Void <u>initDisplayText</u> (byte qualifier, byte dcs, byte[] buffer, short offset, short length)	INDTBB_BSS	000011
Woid <pre>initGetInkey(byte qualifier, byte dcs, byte[]</pre> buffer, short offset, short length)	INGKBB_BSS	000100
Void <pre>initGetInput(byte qualifier, byte dcs, byte[] buffer, short offset, short length, short</pre>	INGPBB_BSSSS	000101
minRespLength, short maxRespLength)	OFNE	000110
Byte send()	SEND	000110
Short getCapacity()	GCAP	011000
Void initCloseChannel(byte bChannelIdentifier)	ICCHB	011001
Inherited Method Name: EditHandler		
Void appendArray(byte[] buffer, short offset, short length, short dstLength)	APDA_BSS	000111
Void appendTLV(byte tag, byte value)	APTLBB	001000
Void appendTLV(byte tag, byte[] value, short valueOffset, short valueLength)	APTLB_BSS	001001
Void appendTLV(byte tag, byte value1, byte value2)	APTLBBB	001010
Void appendTLV(byte tag, byte value1, byte[] value2, short value2Offset, short value2Length)	APTLBB_BSS	001011
Void clear()	CLER	001100
Inherited Method Name: ViewHandler		
Byte compareValue(short valueOffset,byte[] compareBuffer,sh ort compareOffset, short compareLength)	CPRVS_BSS	001101
Short compareoriset, short compared gring short copy(byte[] dstBuffer, short dstOffset, short dstLength)	COPY_BSS	001110
Short copyValue(short valueOffset, byte[] dstBuffer,short dstOffset,short dstLength)	CPYVS_BSS	001111
Byte findAndCompareValue(byte tag,byte[] compareBuffer,shor t compareOffset)	FACRB_BS	010000
Byte findAndCompareValue(byte tag,byte occurence, short valueOffset,byte[] compareBuffer,short compareOffset,short compareLength)	FACRBBS_BSS	010001
Short findAndCopyValue(byte tag,byte occurence,short valueOf fset, byte[] dstBuffer, short dstOffset, short dstLength)	FACYBBS_BSS	010010
Short findAndCopyValue(byte tag,byte[] dstBuffer,short dstOf fset)	FACYB_BS	010011
Byte findTLV(byte tag,byte occurrence)	FINDBB	010100
Short getLength()	GLEN	010101
getLength() Byte qetValueByte(short valueOffset)	GVBYS	010110
Short getValueLength()	GVLE	010111

A.2.8 ProactiveResponseHandler methods

Method Name	Acronyms	Numbering on 6 bits	
Short copyAdditionalInformation(byte[] dstBuffer,	CPAI_BSS	000001	
short dstOffset, short dstLength)	-		
Short <pre>copyTextString</pre> (byte[] dstBuffer, short	CPTS_BS	000010	
dstOffset)			
Short <u>getAdditionalInformationLength</u> ()	GTIL	000011	
Byte getGeneralResult()	GTGR	000100	
Byte getItemIdentifier()	GTII	000101	
Byte getTextStringCodingScheme()	GTCS	000110	
Short getTextStringLength()	GTTL	000111	
ProactiveResponseHandler getTheHandler()	GTHD	001000	
Short getCapacity()	GCAP	010100	
Byte getChannelIdentifier()	GCID	010101	
Short copyChannelData(byte[] dstBuffer,			
short dstOffset, short dstLength)	CCHD_BSS	010110	
short distorriset, short distribution			
Inherited Method Name: ViewHandler		+	
Byte	ODD/(0, D00	004004	
CompareValue(short valueOffset,byte[] compareBuffer,s	CPRVS_BSS	001001	
hort compareOffset, short compareLength)			
Short	COPY_BSS	001010	
Copy(byte[] dstBuffer,short dstOffset,short dstLength	COI 1_B33	001010	
)			
Short	CPYVS_BSS	001011	
CopyValue(short valueOffset,	0 5_200	33.3	
<pre>byte[] dstBuffer,short dstOffset,short dstLength)</pre>			
Byte	FACRB_BS	001100	
FindAndCompareValue(byte tag,byte[] compareBuffer,sho			
rt compareOffset)			
Byte findAndCompareValue(byte tag,byte occurence,	EACREDO DOS	004404	
short valueOffset,byte[] compareBuffer,short compareO	FACRBBS_BSS	001101	
ffset, short compareLength)			
Short.	FACYBBS_BSS	001110	
FindAndCopyValue(byte tag,byte occurence,short value)	1701000_000	001110	
<pre>ffset, byte[] dstBuffer, short dstOffset,</pre>			
short dstLength)			
Short	FACYB_BS	001111	
<pre>findAndCopyValue(byte tag,byte[] dstBuffer,short dst0</pre>			
ffset)			
Byte	FINDBB	010000	
FindTLV(byte tag,byte occurrence)	01.50	0.46.55.	
Short Sationath()	GLEN	010001	
GetLength()	CVDVC	040040	
Byte GetValueByte(short valueOffset)	GVBYS	010010	
Short	GVLE	010011	
GetValueLength()	GVLE	010011	
occvarachenden()			

A.2.9 ToolkitRegistry methods

Method Name	Acronyms	Numbering on 6 bits
AllocateTimer()	ATIM	000001
<pre>changeMenuEntry(byte id, byte[] menuEntry, short offset, short length, byte nextAction, boolean helpSupported, byte iconQualifier, short iconIdentifier)</pre>	CMETB_BSSBZBS	000010
<pre>clearEvent(byte event)</pre>	CEVTB	000011
disableMenuEntry(byte id)	DMETB	000100
<pre>enableMenuEntry(byte id)</pre>	EMETB	000101
<pre>getEntry()</pre>	GETY	000110
<pre>getPollInterval()</pre>	GPOL	000111
<pre>initMenuEntry(byte[] menuEntry, short offset, short length, byte nextAction, boolean helpSupported, byte iconQualifier, short iconIdentifier)</pre>	IMET_BSSBZBS	001000
<pre>isEventSet(byte event)</pre>	IEVSB	001001

Method Name	Acronyms	Numbering on 6 bits
<pre>releaseTimer(byte timerIdentifier)</pre>	RTIM	001010
<u>requestPollInterval</u> (short duration)	RPOL	001011
<pre>setEvent(byte event)</pre>	SEVTB	001100
<pre>setEventList(byte[] eventList, short offset, short</pre>	SEVL_BSS	001101
length)		

A.2.10 ViewHandler methods

The numbering of the ViewHandler methods it will be done in the classes inherit it: EditHandler, EnvelopeHandler, ProactiveResponseHandler, ProactiveHandler, because the methods provided by this class as it is declared 'abstract'.

A.2.11 ToolkitException methods

Method Name	Acronyms	Numbering on 6 bits
Static void throwIt(short reason)	THITS	000001
ToolkitException(short reason)	COORS	000010
Constants	CONS	000011

Annex B (normative): Script file syntax and format description

B.1 Syntax description

Following is a syntax description in BNF.

```
<statement list> ::=
                    [ <statement> \n] +
<statement> ::=
                      <simple> | <switch> | <blank line>
<simple> ::=
                      <reset> | <init> | <command> | <remark>
<reset> ::=
                      RST
<init> ::=
                      INI < hexdata>
<command> ::=
                      CMD < hexdata > [ < response > ] ( < status > )
                      [ < hexdata> ]
<response> ::=
<status> ::=
                      ( < hexdata > )
<remark> ::=
                      REM < text line>
<switch> ::=
                      SWI { [<|abelled list>] + }
<labelled list> ::=
                      <label>: \n <statement list>
```

Description of syntax metalanguage :

\n represents a linebreak

[x] means x can appear optionally

[x] + means 1 or more appearances of x

x | y means x or y

[]{}: (bold) these are characters that appear literally in the script files

<text line> any character until the end of the line <blank line> a line containing no text is acceptable

<hexdata> data written in hexadecimal, each byte separated from the following by a whitespace

Each simple statement beginning with 3 characters different than the ones defined indicates another tool command, and shall be ignored by the parser if not recognized.

```
' ', '\t' : Can be used as separator
```

A long statement can be broken into several lines by using the character '\' at the end of each line which is not the last one in the statement.

For more details refer to the examples in B.3.

B.2 Semantics

Following is the meaning of each of the statements:

Sends an APDU Command to the card, including (optionally) the expected response data and also (optionally) the expected status words SW1, SW2.

RST: Resets and powers on the card

INI: Performs the terminal profile with the following data. Afterwards, it shall perform all the fetch and terminal response commands until there is no proactive session in progress.

REM: Used for comments

SWI: Activates a switch condition. Every labelled list represents a list of statements to be executed, if the label matches the SW resulting from the previously executed command.

Evaluation of expected response and status in the case of a CMD:

<response> data within [...] has to be checked, it needs to be present for an outgoing command. Bytes written as XX
shall not be checked by the APDU tool.

status contained within (...) has to be checked; when several status are valid they shall be separated by commas. Nibble written as X shall not be checked by the APDU tool.

B.3 Example

```
REM this is an example
RST
REM Case 1 example
CMD A0 C2 00 00 00 (91 33 , 69 XX)
REM Case 2 example
CMD A0 B6 00 00 07
   [XX XX XX 55 55 XX 55] \
    (91 33 , 67 XX)
CMD A0 B6 00 00 07 \
   (91 33 , 67 XX)
CMD A0 C0 00 00 1F \
   [10 A0 00 00 00 09 00 02 FF FF FF FF 89 28 A4 05 \
   02 0D CC | \
    (90 00)
REM Case 3 example
CMD A0 C2 00 00 33 \
   D1 31 82 02 83 81 06 05 80 11 22 33 44 8B 24 40 \
   08 00 24 23 85 18 41 04 51 10 10 00 00 00 00 13 \
   02 70 00 00 0E 0D 00 00 00 00 28 A4 05 00 00 00 \
   00 00 00 \
   (90 00)
REM Case 4 example with switch statement
CMD 00 A4 04 00 10 \
   A0 00 00 00 09 00 02 FF FF FF FF 89 41 04 44 02 \backslash
   (61 XX, 6A 82)
SWI {
61 XX:
CMD 00 C0 00 00 14 \
   [10 A0 00 00 00 09 00 02 FF FF FF FF 89 41 04 44 \
   02 02 CC CC] \
   (90 00)
CMD A0 A4 00 00 02 \
   3F 00
6A 82:
RST
```

```
REM Case 5 example
CMD A0 C2 00 00 33 \
    D1 31 82 02 83 81 06 05 80 11 22 33 44 8B 24 40 \
    08 00 24 23 85 18 41 04 51 10 10 00 00 00 00 13 \
    02 70 00 00 0E 0D 00 00 00 28 A4 05 00 00 00 \
    (6X 00)
```

B.4 Style and formatting

In order to show a common appearance all the scripts shall follow those format rules:

- start always with a 'RST'.
- The command, data to be checked and status to be checked shall be presented in the following order:

```
CMD COMMAND [EXPECTED DATA] (EXPECTED STATUS)
```

- APDU shall be presented with command (CLA INS P1 P2 P3) in one line and data (if present) in next line grouped 16 bytes per line (see example above).
- The expected data (if present) shall be presented in 16 bytes groups per line (see example above).

Annex C (normative): Default Prepersonalization

C.1 General Default Prepersonalization

This table shows the default prepersonalization, the file system and the files' content, that the test SIM cards shall contain unless otherwise stated.

Name	Identifier	Default Value	Special Features
EFICCID	2FE2	OF FF FF FF FF FF FF FF	This value is not compliant with 3GPP TS 51.011 [3]
EF _{IMSI}	6F07	FF FF FF FF FF FF FF	This value is not compliant with 3GPP TS 51.011 [3]
EF _{LP}	6F05	01 FF FF FF	
EF _{Kc}	6F20	FF FF FF FF FF FF FF 07	
EF _{PLMNsel}	6F30	FF	
		FF	
EF _{HPLMN}	6F31	05	
EF _{ACMmax}	6F37	00 00 00	Access condition UPDATE: CHV1
EF _{SST}	6F38	FF 3F C3 0F 0C 00 FF 0F 00 33	
EF _{ACM}	6F39	00 00 00	Access condition UPDATE: CHV1
EF _{PUCT}	6F41	FF FF FF 00 00	Access condition UPDATE: CHV1
EF _{BCCH}	6F74	FF	
EE	6570	FF FF FF FF 00 00	
EF _{ACC} EF _{FPLMN}	6F78 6F7B	FF	
	6F7E	FF FF FF FF 00 F0 00 00 00 FF 01	
EFLOCI		00 FF FF	
EF _{AD}	6FAD 6FAE	03	
EF _{Phase}	6F3B	Default value in all the records:	Records: 5
EF _{FDN}	0530	FF	Records. 5
		FF FF FF FF FF FF FF FF FF	
		FF FF FF FF	
EF _{SMSP}	6F42	FF	Records: 1
		FF	
		FF	
EF _{LND}	6F44	FF	Records: 1
E. LND	0	FF	Treasure I
		FF FF FF FF	
EF _{SMSS}	6F43	FF FF	
EF _{SMS}	6F3C	1st record: 00 FF FF(length 176)	Records: 3
		2 nd record:00 FF FF(length 176) 3 rd record: 00 FF FF(length 176)	
EF _{ADN}	6F3A	FF	Records: 1
LIADN	01 0/1	FF FF FF FF FF FF FF FF FF	Trecords. 1
		FF FF FF FF	
EF _{CCP}	6F3D	FF	
FF	0540	FF	December 4
EF _{MSISDN}	6F40	FF	Records: 1
		FF FF FF	
EF _{SDN}	6F49	FF	Records: 1
		FF	
	055.4	FF FF FF FF	
EF _{SUME}	6F54	85 0C 54 4F 4F 4C 4B 49 54 20 54 45 53 54 FF FF FF FF	
EF _{CBMI}	6F45	FF FF	
EF _{CBMID}	6F48	10 80	
EF _{CBMIR}	6F50	10 80 10 9F	
EF _{IM}	4F20	FF	
LI IIVI	T1 40		

The default value for the CHV1 shall be " $0x31\ 0x31\ 0x31\ 0x31\ 0xFF\ 0xFF\ 0xFF\ 0xFF$ " and its state shall be 'disabled' during test applets execution.

C.2 Sim.Access.SimView test default prepersonalization

C.2.1 DF_{SIMTEST} (SIM Test)

Identifier: '0319'

C.2.2 EF_{TNR} (Transparent Never Read)

	Identifier: '6F01'		Structure: transparent		ndatory
	File size: 3 bytes		Update activi	ty: low	
Access Conditions:					
	READ		NEVER		
	UPDATE		ALWAYS		
	INVALI	DATE	ALWAYS		
	REHA	BILITATE	ALWAYS		
Bytes	Description		Default Value	M/O	Length
1 - 3	Test Data		AA AA AA	М	3 bytes

C.2.3 EF_{TNU} (Transparent Never Update)

Identifier: '6F02' Stru		ructure: transparent Mandato		ndatory	
File size: 3 bytes		Update activity: low			
Access Conditions:					
	READ		ALWAYS		
	UPDATE		NEVER		
	INVALIC	ATE	ALWAYS		
	REHABI	LITATE	ALWAYS		
Bytes	Description	Default Value		M/O	Length
1 - 3	Test Data		55 55 55	М	3 bytes

C.2.4 EF_{TARU} (Transparent Always Read and Update)

Identifier: '6F03'		tructure: transparent Mandatory		andatory	
File size: 260 bytes			Update activity	: low	
		ons:			
	READ		ALWAYS		
	UPDATE		ALWAYS		
	INVALID	ATE	ALWAYS		
	REHABI	LITATE	ALWAYS		
Bytes	Description		efault Value	M/O	Length
1 - 260	Test Data	<u> </u>	FF FF	М	260 bytes

C.2.5 EF_{CNR} (Cyclic Never Read)

Iden	tifier: '6F04'	S	Structure: cyclic		Mandatory
R	ecord length: 3 bytes		Update	e activity:	high
	A READ	ccess Con	ditions:	R	
	UPDATE INCREA	SE	ALWA)	/S /S	
	INVALID REHABI		ALWA\	_	
Logical Record Number	Description		Default Value	M/O	Length
1	Test Data		00 00 00	М	3 bytes
2	Test Data		00 00 00	М	3 bytes

C.2.6 EF_{CNU} (Cyclic Never Update)

Iden	tifier: '6F05'	· ·	Structure: cyclic		Mandatory		
R	ecord length: 3 bytes		Updat	e activity:	high		
	Access Conditions:						
	READ UPDA	TE	ALWA` NEVE				
	INCRI INVAL		NEVE ALWA				
	REHA	BILITATE	ALWA	YS			
Logical Record Number	Description		Default Value	M/O	Length		
1	Test Data		00 00 00	М	3 bytes		
2	Test Data		00 00 00	М	3 bytes		

C.2.7 EF_{CNIC} (Cyclic Never Increase)

Iden	tifier: '6F06	;	Structure: cyclic		Mandatory		
R	Record length: 3 bytes		Updat	e activity	: high		
	Access Conditions: READ ALWAYS						
	READ UPDATE			YS			
	INCRI INVAL		NEVE ALWA				
	REHAI	BILITATE	ALWA	YS			
Logical Record Number	Description		Default Value	M/O	Length		
1	Test Data		00 00 00	М	3 bytes		
2	Test Data		00 00 00	М	3 bytes		

C.2.8 EF_{CNIV} (Cyclic Never Invalidate)

Iden	tifier: '6F07	;	Structure: cyclic		Mandatory
R	ecord length: 3 bytes		Upda	te activity	: high
Access Conditions: READ ALWAYS UPDATE ALWAYS INCREASE ALWAYS INVALIDATE NEVER REHABILITATE ALWAYS					
Logical Record Number	Description		Default Value	M/O	Length
1	Test Data		00 00 00	М	3 bytes
2	Test Data		00 00 00	М	3 bytes

C.2.9 EF_{CNRH} (Cyclic Never Rehabilitate)

Iden	tifier: '6F08'	;	Structure: cyclic		Mandatory		
R	ecord length: 3 bytes		Updat	e activity:	high		
	Access Conditions:						
	READ UPDAT	Έ	ALWA` ALWA`				
	INCRE INVALI		ALWA` ALWA`				
	REHA	BILITATE	NEVE	R			
Logical Record Number	Description		Default Value	M/O	Length		
1	Test Data		00 00 00	М	3 bytes		
2	Test Data		00 00 00	М	3 bytes		

C.2.10 EF_{CARU} (Cyclic Always Read and Update)

Ident	tifier: '6F09'	;	Structure: cyclic		Mandatory			
R	ecord length: 3 bytes		Updat	e activity:	high			
	Access Conditions:							
	READ		ALWA'	YS				
	UPDATE		ALWA'	YS				
	INCREASE		ALWA'	YS				
	INVALIDATE	•	ALWA'	YS				
	REHABILIT <i>A</i>	TE	ALWA'	YS				
Logical	Description		Default Value	M/O	Length			
Record					Ü			
Number								
1	Test Data		55 55 55	М	3 bytes			
2	Test Data		AA AA AA	М	3 bytes			

C.2.11 EF_{LNR} (Linear Fixed Never Read)

	Identifier: '6F0A'	Str	Structure: linear fixed Mandatory		ndatory	
Record length: 4 bytes			Update activity: low			
	Access Conditions:					
	READ		NEVER			
	UPDATE	<u> </u>	ALWAYS			
	INVALID	ATE	ALWAYS			
	REHABI	LITATE	ALWAYS			
Logical	Description		Default Value	M/O	Length	
Record	•					
Number						
1	Test Data - Record 1		FF FF FF FF	M	4 bytes	
2	Test Data - Record 2		FF FF FF FF	М	4 bytes	

C.2.12 EF_{LNU} (Linear Fixed Never Update)

	Identifier: '6F0B'		ucture: linear fixed	Mar	ndatory
Record length: 4 bytes			Update activity: low		
	Access Conditions:				
	READ		ALWAYS		
	UPDATE	Ī	NEVER		
	INVALIDA	ATE	ALWAYS		
	REHABIL	LITATE	ALWAYS		
Logical	Description		Default Value	M/O	Length
Record	2 00011,511011		z oraan varae	1, 0	_0g
Number					
1	Test Data - Record 1		FF FF FF FF	М	4 bytes
2	Test Data - Record 2		FF FF FF FF	М	4 bytes

C.2.13 EF_{LARU} (Linear Fixed Always Read and Update)

Identifier: '6F0C'		Str	Structure: linear fixed Mandate		ndatory	
Record length: 4 bytes			Update activity: low			
Access Conditions:						
	READ UPDATE INVALID REHABI	ATE	ALWAYS ALWAYS ALWAYS ALWAYS			
Logical Record Number	Description		Default Value	M/O	Length	
1	Test Data - Record 1		55 55 55 55	М	4 bytes	
2	Test Data - Record 2		AA AA AA AA	М	4 bytes	

C.2.14 EF_{CINA} (Cyclic Increase Not Allowed)

Identifier:	Identifier: '6F0D'		Man	datory
Record	Record length: 3 bytes		: high	
	Access Co	onditions:		
	READ	ALWAYS		
	UPDATE	ALWAYS		
	INCREASE	ALWAYS (see note)		
	INVALIDATE	ALWAYS		
	REHABILITATE	ALWAYS		
Logical Record Number	Description	Default Value	M/O	Length
1	Test Data	00 00 00	М	3 bytes
2	Test Data	00 00 00	М	3 bytes
		h that increase is not allowed, as Il structure of an EF returned by		

C.2.15 EF_{TRAC} (Transparent Read Access Condition CHV2)

Identifier: '6F0E'		Stru	ucture: transparent	Man	datory
Reco	ord length: 3 bytes		Update activ	vity: low	
	Ad	cess Condit	ions:		
	READ		CHV2		
UPDATE			ALWAYS		
INCREASE		SE	ALWAYS		
INVALIDATE		ATE	ALWAYS		
REHABILIT		.ITATE	ALWAYS		
Logical Record Number	Description		Default Value	M/O	Length
1	Test Data		00 00 00	М	3 bytes

C.2.16 EF_{TIAC} (Transparent Invalidate Access Condition CHV1)

Identifier: '6F0F'		Str	ucture: transparent	Man	datory
Record length: 3 bytes		Update activi	ty: low		
	Ad	ccess Condit	ions:		
	READ		ALWAYS		
UPDATE		ALWAYS			
	INCREASE		ALWAYS		
	INVALIDATE		CHV1		
	REHABIL	.ITATE	ALWAYS		
Logical Record	Description		Default Value	M/O	Length
Number					
1	Test Data		00 00 00	М	3 bytes

C.2.17 EF_{CIAC} (Cyclic Increase Access Condition CHV2)

Identifier: '6F10'			Structure: cyclic		Man	datory
Reco	Record length: 3 bytes		Update a	ctivity:	low	
Access Condi		ccess Condit	ions:			
	READ		ALWAYS			
	UPDATE		ALWAYS			
	INCREASE		CHV2			
	INVALIDATE		ALWAYS			
	REHABIL	ITATE	ALWAYS			
Lasical Decard	Description		Default Value		14/0	l a sa astla
Logical Record	Description		Default Value		M/O	Length
Number						
1	Test Data		00 00 00		M	3 bytes
2	Test Data		00 00 00		М	3 bytes

C.2.18 EF_{CIAA} (Cyclic Increase Access Condition ADM)

Identifier: '6F11'		;	Structure: cyclic		Man	datory
Reco	Record length: 3 bytes		Update activity: low			
Access Conditions:						
	READ		ALWAYS			
	UPDATE		ALWAYS			
	INCREASE		ADM			
	INVALIDATE		ALWAYS			
REHABILITATE		ITATE	ALWAYS			
						,
Logical Record	Description		Default Value		M/O	Length
Number						
1	Test Data		00 00 00		М	3 bytes
2	Test Data		00 00 00	•	М	3 bytes

C.2.19 EF_{CNRI} (Cyclic Never Rehabilitate Invalidated)

Identifie	er: '6F12'		Structure: cyclic	Mai	ndatory
Record length: 3 bytes		Update act	ivity: low		
Access Conditions:					
	READ		ALWAYS		
	UPDATE		ALWAYS		
	INCREASE		ALWAYS		
	INVALIDATE		ALWAYS		
	REHABI	LITATE	NEVER		
Logical Pocard	Description		Default Value	M/O	Longth
Logical Record Number	Description		Delault Value	IVI/O	Length
1	Test Data		00 00 00	M	3 bytes
2	Test Data		00 00 00	M	3 bytes

The file status shall be invalidated as defined in 3GPP TS 51.011 [3].

Annex D (normative): sim.test.util package and loading, testing and cleaning script examples

See attached files:

- Annex_D_SimTestUtil.zip
- Annex_D_Examples.zip

Annex E (normative): Test Area files

See attached file:

- Annex_E_SourceCode.zip

Annex F (normative): AID numbering and acronyms for Framework tests

F.1 Toolkit Installation Parameters (TIN)

Test Area within the chapter	Acronyms	Numbering on 6 bits
Timer allocation	TMAL	000001
Item identifier	ITID	000010
Item position	ITPO	000011
Access conditions	ACCO	000100
Priority level	PRLV	000101
Maximum length for each menu entry	MLME	000110
Number of menu entries	NBME	000111
Memory space	MESP	001000
Channel Allocation	CHAL	001001
Minimum Security Level	MSL	001010

F.2 Minimum Handler Availability (MHA)

Test Area within the chapter	Acronyms	Numbering on 6 bits
ProactiveHandler	PAHD	000001
ProactiveResponseHandler	PRHD	000010
EnvelopeHandler	ENHD	000011
EnvelopeResponseHandler	ERHD	000100

F.3 Handler Integrity (HIN)

Test Area within the chapter	Acronyms	Numbering on 6 bits
ProactiveHandler	PAHD	000001
ProactiveResponseHandler	PRHD	000010
EnvelopeHandler	ENHD	000011
EnvelopeResponseHandler	ERHD	000100

F.4 Applet Triggering (APT)

Test Area within the chapter	Acronyms	Numbering on 6 bits
EVENT_PROFILE_DOWNLOAD	EPDW	000001
EVENT_MENU_SELECTION	EMSE	000010
EVENT_MENU_SELECTION_HELP_REQUEST	EMSH	000011
EVENT_FORMATTED_SMS_PP_ENV	EFSE	000100
EVENT_UNFORMATTED_SMS_PP_ENV	EUSE	000101
EVENT_CALL_CONTROL_BY_SIM	ECCN	000110
EVENT_MO_SHORT_MESSAGE_CONTROL_BY_SIM	EMCN	000111
EVENT_TIMER_EXPIRATION	ETEX	001000
EVENT_UNFORMATTED_SMS_CB	EUCB	001001
EVENT_EVENT_DOWNLOAD_MT_CALL	EDMC	001010
EVENT_EVENT_DOWNLOAD_CALL_CONNECTED	EDCC	001011
EVENT_EVENT_DOWNLOAD_CALL_DISCONNECTED	EDCD	001100
EVENT_EVENT_DOWNLOAD_LOCATION_STATUS	EDLS	001101
EVENT_EVENT_DOWNLOAD_USER_ACTIVITY	EDUA	001110
EVENT_EVENT_DOWNLOAD_IDLE_SCREEN_AVAILABLE	EDIS	001111

Test Area within the chapter	Acronyms	Numbering on 6 bits
EVENT_EVENT_DOWNLOAD_CARD_READER_STATUS	EDCR	010000
EVENT_UNRECOGNIZED_ENVELOPE	EUEV	010001
EVENT_STATUS_COMMAND	ESTC	010010
EVENT_EVENT_DOWNLOAD_LANGUAGE_SELECTION	EDLG	010011
EVENT_EVENT_DOWNLOAD_BROWSER_TERMINATION	EDBT	010100
EVENT_FORMATTED_SMS_CB	EFCB	010101
EVENT_FIRST_COMMAND_AFTER_SELECT	EFCA	010110
EVENT_EVENT_DOWNLOAD_DATA_AVAILABLE	EDDA	010111
EVENT_EVENT_DOWNLOAD_CHANNEL_STATUS	EDCS	011000
EVENT_FORMATTED_SMS_PP_UPD	EFSU	011001
EVENT_UNFORMATTED_SMS_PP_UPD	EUSU	011010

F.5 Proactive Command Sending (PCS)

Test Area within the chapter	Acronyms	Numbering on 6 bits
System Proactive commands	SPCO	000001
Interaction with GSM commands	IGCO	000010
Errors during proactive command sending	EPCS	000011
Proactive Command Control	PCCO	000100

F.6 Envelope Response Posting (ERP)

Test Area within the chapter	Acronyms	Numbering on 6 bits
EVENT_CALL_CONTROL_BY_SIM	ECCN	000001
EVENT_MO_SHORT_MESSAGE_CONTROL_BY_SIM	EMCN	000010
EVENT_UNRECOGNIZED_ENVELOPE	EUEN	000011
EVENT_FORMATTED_SMS_PP_ENV	EFSE	000010

F.7 Framework Security (FWS)

Test Area within the chapter	Acronyms	Numbering on 6 bits	
Input data	INDA	000001	
Output data	OUDA	000010	

F.8 File System Context (FSC)

Test Area within the chapter	Acronyms	Numbering on 6 bits
Initial Context	INIT	000001
Context Preservation for Current File	CUFI	000010
Context Preservation for Current Record	CURE	000011

F.9 Exception Handling (EXH)

Test Area within the chapter	Acronyms	Numbering on 6 bits
Hide exception to the mobile	HEME	000001
Interaction with multi-triggering	IMTG	000010

F.10 Other parts transferred to framework from API (API)

Test Area within the chapter	Acronyms	Numbering on 6 bits		
A handler is a temporary JCRE Entry Point object	HEPO	000001		
Transaction	TRAN	000010		
Timer Id between Applets	TMID	000011		

F.11 Concatenation processing (PROC)

Test Area within the chapter	Acronyms	Numbering on 6 bits
Concatenation processing	PROC	000001

Annex G (normative): Configuration Parameters File

This file describes all the mandatory and optional parameters that are used in order to create the loading script(s) for one test area. The configuration parameters file contains the values for the parameters needed in order to generate the loading and cleanup scripts.

The name of the parameters file will be *<test area reference>_<n>.par*.

The number <n> is associated with the loading/cleanup script number, i.e. API_2_TKR_ SEVL_BSS_1.par is used to generate API_2_TKR_ SEVL_BSS_1.ldr etc.

G.1 Syntax

The general syntax for this file will be:

```
<file> ::= <section>+
<section> ::= <section heading> <line break> <section body>
<section heading> ::= '[' <name> ']'
<section body> ::= <parameter assignment>+
<parameter assignment> ::= <name> '=' <value> <line break></parameter
```

Where '+' indicates one or more repetitions of the previous syntax element.

Any text included between the symbol ';' and the end of line is considered a comment and ignored by parsing tools.

Empty values are considered valid. They are used to indicate that an optional value is not present.

Names of sections, names of parameters and values are case-sensitive.

Blank spaces and Tabs between tokens are allowed and will be ignored by the parser.

When values represent a sequence of bytes, they are expressed in hexadecimal format, where every 2 digits represent one byte. Blank space between bytes is optional.

Example:

```
; comment

[Section1]

Parameter11 = 00 11 22 33

Parameter12 = 0101  ; another comment

[Section2]

Parameter21 = vvwwxxyyzz
```

G.2 File Contents and Organization

Parameters in this file are organized in the following sections:

[CONVERT]	Conversion parameters used during conversion (i.e. CAP file generation)
[INSTALL(load)]	Parameters used by the Install for Load command
[LOAD]	Parameters used by the Load command
[INSTALL(install)]	Parameters used by the Install for Install command

All sections may appear only once in the file, except for the "INSTALL(install)" section. If that section appears more than once, it will apply to different applet instances, in sequence.

G.2.1 Default values, order and processing

The ordering of the parameters and the sections is relevant, since parameter names may be repeated and apply to different applets.

When one single parameter is repeated within one section, it refers to different applets. The value of the n^{th} appearance of the parameter applies to applet n.

When one section is repeated (INSTALL(install)), then the n^{th} appearance of the section applies to applet n. Parameter/value pairs which are found in one appearance of the section are valid for the subsequent applets as long as they are not overridden. For example, first INSTALL(install) may contain all values for parameters, whereas the subsequent INSTALL(install) sections may only contain parameters whose values change.

If one required parameter is missing from one section, the last defined value of this parameter in a previous section of the same file will be used.

G.2.2 CONVERT Section

These parameters allow configuration of the conversion process of the Java class file(s) into one CAP file.

Parameter	Description
PackageAID	AID of the package
PackageName	Fully qualified name of the package
PackageVersion	Version of the package
AppletClassAID	AID of the applet
AppletClassName	Name of the applet

G.2.3 INSTALL(load) Section

Here are the parameters to be included in the Install(Load) command (as specified in TS 23.048 [8]).

Parameter	Description
PackageAID	AID of the package
PackageNonVolatileMemSize	Non Volatile memory space (in bytes) required for package loading
InstallationNonVolatileMemSize	Non volatile memory required for installation, in bytes
InstallationVolatileMemSize	Volatile memory required for installation, in bytes

G.2.4 LOAD Section

Here are the parameters to be included in the Load command (as specified in TS 23.048 [8]).

Parameter	Description
MaxLoadCommandDataLength	Maximum length of the data provided in the load command (P3
	parameter of the LOAD APDU embedded in the command packet)

G.2.5 INSTALL(install) Section

Here are the parameters to be included in the Install(Install) command (as specified in 3GPP TS 23.048 [8]).

Parameter	Description
PackageAID	AID of the package
AppletClassAID	AID of the applet
InstanceAID	AID of the instance of the applet
InstallationNonVolatileMemSize	Non volatile memory required for installation, in bytes
InstallationVolatileMemSize	Volatile memory required for installation, in bytes
AccessDomain	Specify the SIM files that may be accessed by the applet and the operations allowed on these files. This parameter includes the Access Domain Parameter (ADP) and Access Domain Data (ADD)
PriorityLevel	Priority level of the Toolkit applet instance
MaxNumberOfTimers	Maximum number of timers allowed for this applet instance
MaxMenuEntryTextLength	Maximum text length for a menu entry
MaxNumberOfMenuEntries	Maximum number of menu entries allowed for this applet instance
MenuEntriesPositionIdentifier	For each menu entry: Position and identifier of that menu entry
MaxNumberOfChannels	Maximum Number of channels for this applet instance
MSLFieldLength	Length of Minimum Security Level field
MSLParameter	MSL Parameter
MSLData	MSL Data
AppletSpecificParameters	Parameters specific to the applet

The applet shall be installed with install(install and make selectable) command.

G.3 Full example

```
[CONVERT]
PackageAID = A0 00 00 00 30 00 02 FF FF FF FF 89 00 00 01 00
PackageName = sim.test.access.api_1_svw_updrbs
PackageVersion = 1.0
AppletClassAID = A0 00 00 00 30 00 02 FF FF FF FF 89 00 00 01 01
AppletClassName = API_1_SVW_UPDRBS_1
AppletClassAID = A0 00 00 00 30 00 02 FF FF FF FF 89 00 00 01 02
AppletClassName = API_1_SVW_UPDRBS_2
[INSTALL(load)]
PackageNonVolatileMemSize = 0D27
;InstallationNonVolatileMemSize = 0400
;InstallationVolatileMemSize = 0000
[LOAD]
MaxLoadCommandDataLength = 6C ; max value
[INSTALL(install)]
AppletClassAID = A0 00 00 00 30 00 02 FF FF FF FF 89 00 00 01 01
```

```
InstanceAID = A0 00 00 00 30 00 02 FF FF FF FF 89 00 00 01 01
InstallationNonVolatileMemSize = 0400
InstallationVolatileMemSize = 0000
AccessDomain = 00
PriorityLevel = FF
MaxNumberOfTimers = 00
MaxMenuEntryTextLength = 10
MaxNumberOfMenuEntries = 01
MenuEntriesPositionIdentifier = 0001
AppletSpecificParameters =
[INSTALL(install)]
AppletClassAID = A0 00 00 00 30 00 02 FF FF FF FF 89 00 00 01 02
InstanceAID = A0 00 00 00 30 00 02 FF FF FF FF 89 00 00 01 02
InstallationNonVolatileMemSize = 0200
InstallationVolatileMemSize = 0000
MenuEntriesPositionIdentifier = 0002
MaxNumberOfChannels = 05
MSLFieldLength = 00
MSLParameter =
MSLData =
```

[;] rest of INSTALL(install) parameters are taken from previous INSTALL(install)...

Annex H (informative): Change history

The table below indicates all changes that have been made to the present document since drafting work began.

Change history								
Date	TSG#	TSG Doc	CR	Rev	Cat	Subject/Comment	Old	New
2000-10	-	-				Draft presented at T3 #16		0.2.0
2000-12	TP-10	TP-000208				Presented to TSG-T #10 for information	0.2.0	1.0.0
2001-01	-	-				Input to T3 #17 resulting from T3 ad hoc #24	1.0.0	1.1.0
2001-03	-	-				Document presented for approval at T3 #18	1.1.0	1.2.0
2001-03	TP-11	TP-010041				Document presented for approval to TSG-T #11 (identical in technical content to v1.2.0)	1.2.0	2.0.0
2001-03						As approved at TSG-T #11 (identical in technical content to v2.0.0)	2.0.0	7.0.0
2001-05						Correction to date on cover page / headers	7.0.0	7.0.1
2001-06	TP-12	TP-010105	A001	-	F	Corrections to the API Test plan, addition of the test area files and modification of the until package	7.0.1	7.1.0
2001-09	TP-13	TP-010206	A002	-	F	Update API Test plan and Test Area Files	7.1.0	7.2.0
2001-11	TP-14	TP-010241	A003	-	F	Specification for framework part	7.2.0	7.3.0
			A004	-	F	API part		
2002-03	TP-15	TP-020073	004	-	F	Testing Framework Update	7.3.0	7.4.0
2002-03						Files for Annexes D and E added, Editorial correction performed in 6.3.2.3.3 column 9 (1- Applet triggered)	7.4.0	7.4.1
2002-09	T3#24					Reference [14] changed as TS 101 220 v3.0.0 was withdrawn.	7.4.1	7.4.2
2002-09	TP-17					Specification upgraded to release 99 without any changes. The technical content is identical to the previous version 7.4.2	7.4.2	8.0.0
2002-12	TP-18	TP-020285	A005	-	F	Update of 11.13 Specification for Release 99	8.0.0	8.1.0
2003-03	TP-19	TP-030026	A006	-	F	Corrections on 11.13 Specification	8.1.0	4.0.0
			A007	-	F	Upgrade of 11.13 Specification to Release 4		
2003-04						Editorial modification: replacement of Annex E source code file.	4.0.0	4.0.1
2003-06	TP-20	TP-030125	001	-	В	Update of 51.013 Specification for Release 5	4.0.1	5.0.0
	_					editorial: replacment of annex E due to problems in the folder-structure.	5.0.0	5.0.1

History

Document history		
V5.0.1	June 2003	Publication