3GPP TSG-T plenary meeting #19 ***Tdoc TP-030026***
Birmingham, UK, 12-14 March 2003

Source: T3

Title: CRs to TS 11.13: Test specification for the SIM API for Java Card™

Document for: Approval

This document contains the following change requests:

T3-Doc	Spec	CR	Rev	Cat	Phase	Subject	Version-Current	Version-New	WI
T3-030168	11.13	A006	-	F	R99	Corrections on 11.13 Specification	8.1.0	8.2.0	TEI
T3-030169	11.13	A007	-	F	Rel-4	Upgrade of 11.13 Specification to Release 4	8.1.0	4.0.0	TEI

3GPP TSG-T3 Meeting #26
Lisbon, Portugal, 11-14 February 2003

Tdoc ⌘T3-030168
Revised T3-030009

CR-Form-v7

CHANGE REQUEST

⌘ **11.13** **CR** **A006** ⌘**rev** **-** ⌘ Current version: **8.1.0** ⌘

*For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.*

Proposed change affects: UICC apps⌘ [X] ME [ ] Radio Access Network [ ] Core Network [ ]

Title:	⌘	Corrections on 11.13 Specificaction
Source:	⌘	T3
Work item code:	⌘	TEI
Date:	⌘	13/02/2003
Category:	⌘	**F**
Release:	⌘	R99

Use one of the following categories:
F *(correction)*
A *(corresponds to a correction in an earlier release)*
B *(addition of feature),*
C *(functional modification of feature)*
D *(editorial modification)*
Detailed explanations of the above categories can be found in 3GPP TR 21.900.

Use one of the following releases:

2	*(GSM Phase 2)*
R96	*(Release 1996)*
R97	*(Release 1997)*
R98	*(Release 1998)*
R99	*(Release 1999)*
Rel-4	*(Release 4)*
Rel-5	*(Release 5)*
Rel-6	*(Release 6)*

Reason for change: ⌘ Corrections in test area files (Annex E) in accordance to a previous correction in the specification.
Coherence in the test format description.

Summary of change: ⌘

- §6.2.2.1.1: Description of new events Envelope Event Download – Language Selection and Envelope Event Download – Browser Termination added.
- §6.2.4.11.2: name of test files corrected.
- §6.2.11.3.2: API_2_TKE_THITS_1.par file added.
- §6.3.1.3.3: Test Procedure corrected according to the format of the rest of the tests
- §6.3.1.3.4: Removal of the second row of the Test Coverage Table.
- §6.3.3.11.2: FWK_APT_EDCC_1.par file added.
- §6.3.6.1.3: Test Procedure corrected according to the format of the rest of the tests
- Annex E, API_2_TKR_RPOLS: new tests, in accordance with spec.
- Annex E, API_2_ENH_GVBYS: sequence of code in java file corrected
- Annex E, API_2_PAH_FACRBBS_BSS: file name corrected
- Annex E, API_2_TKR_CMETB_BSSBZBS: path name corrected

Consequences if not approved: ⌘ Tests are not in accordance with specification.
Inconsistency in the test format description within the specification

Clauses affected:	⌘	6.2.2.1.1, 6.2.4.11.2, 6.2.11.3.2, 6.3.1.3.3, 6.3.3.11.2, 6.3.6.1.3, Annex E			
		Y	**N**		
Other specs affected:	⌘		**X**	Other core specifications	⌘
			X	Test specifications	
			X	O&M Specifications	
Other comments:	⌘				

How to create CRs using this form:

Comprehensive information and tips about how to create CRs can be found at http://www.3gpp.org/specs/CR.htm. Below is a brief summary:

1) Fill out the above form. The symbols above marked ⌘ contain pop-up help information about the field that they are closest to.

2) Obtain the latest version for the release of the specification to which the change is proposed. Use the MS Word "revision marks" feature (also known as "track changes") when making the changes. All 3GPP specifications can be downloaded from the 3GPP server under ftp://ftp.3gpp.org/specs/ For the latest version, look for the directory name with the latest date e.g. 2001-03 contains the specifications resulting from the March 2001 TSG meetings.

3) With "track changes" disabled, paste the entire CR form (use CTRL-A to select it) into the specification just in front of the clause containing the first piece of changed text. Delete those parts of the specification which are not relevant to the change request.

6 API Test Plan

6.2.2 Interface ToolkitInterface

6.2.2.1.1 Conformance Requirement:

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

```
public void processToolkit(byte event)
        throws  ToolkitException
```

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_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_STATUS_COMMAND	Status APDU command event
EVENT_UNRECOGNIZED_ENVELOPE	Unrecognized Envelope command reception

Parameters error

No requirements

Context errors

No requirements

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_l.dr
Cleanup Script:	API_2_ENH_GVBYS_1.clr
Parameter File:	API_2_ENH_GVBYS_1.par

6.2.11.3.2 Test suite files

No additional requirements for the GSM personalisation:

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

6.3.1.3.3 Test Procedure

Id	Description	API/Framework– Expectation	APDU Expectation
1	**Applet1 and Applet2 registration and Envelope Handler availability with EVENT_PROFILE_DOWNLOAD** 1- Applet1 is registered to all events defined [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. Applet2 is registered to all events defined [7] except 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 EVENT_TIMER_EXPIRATION and setEventList for the rest of the events. 2-Terminal Profile command is sent to SIM without the facility of SET_EVENT_LIST ,SETUP_IDLE_MODE_TEXT ,POLL_INTERVAL and SETUP MENU 3-EnvelopeHandler.getTheHandler() method is called by Applet1 Applet1 is deregistered to EVENT_PROFILE_DOWNLOAD 4-EnvelopeHandler.getTheHandler() method is called by Applet2 Applet2 is deregistered to EVENT_PROFILE_DOWNLOAD	1- No exception is thrown 2- Applet1 is triggered 3- A Toolkit exception HANDLER_NOT_AVAILABLE is thrown 4- Applet2 is triggered 5- A Toolkit exception HANDLER_NOT_AVAILABLE is thrown	
2	**Envelope Handler availability with EVENT_MENU_SELECTION_HELP_REQUEST** Perform SIM initialization with all the facilities supported Envelope menu selection with help request is sent to the SIM 1-EnvelopeHandler.getTheHandler() method is called by Applet1 2-Envelope menu selection with help request is sent to the SIM 3-EnvelopeHandler.getTheHandler() method is called by Applet2	1- Applet1 is triggered 2- No exception is thrown. ~~Applet1 finalizes.~~ Applet1 finalizes 3- Applet2 is triggered 4- No exception is thrown.	
3	**Envelope Handler availability with EVENT_MENU_SELECTION** 1-Envelope menu selection is sent to the SIM	1- Applet1 is triggered	

Id	Description	API/Framework– Expectation	APDU Expectation
	2-EnvelopeHandler.getTheHandler() method is called by Applet1 ~~Applet1 finalizes.~~ 3-Envelope menu selection is sent to the SIM 4-EnvelopeHandler.getTheHandler() method is called by Applet2	2- No exception is thrown. Applet1 finalizes 3- Applet2 is triggered 4- No exception is thrown.	

Id	Description	API/Framework– Expectation	APDU Expectation
4	**Envelope Handler availability with EVENT_FORMATTED_SMS_PP_ENV** 1-A EVENT_FORMATTED_SMS_PP_ENV envelope is sent to the SIM 2-EnvelopeHandler.getTheHandler() method is called by Applet1 ~~Applet1 finalizes.~~ 3-A EVENT_FORMATTED_SMS_PP_ENV envelope is sent to the SIM 4-EnvelopeHandler.getTheHandler() method is called by Applet2	1- Applet1 is triggered 2- No exception is thrown. Applet1 finalizes 3- Applet2 is triggered 4- No exception is thrown.	
5	**Envelope Handler availability with EVENT_UNFORMATTED_SMS_PP_ENV** 1-An unformatted sms pp envelope is sent to the SIM 2-EnvelopeHandler.getTheHandler() method is called by Applet1 ~~Applet1 finalizes.~~ 3-EnvelopeHandler.getTheHandler() method is called by Applet2	1- Applet1 is triggered 2- No exception is thrown. Applet1 finalizes 3- Applet2 is triggered 4- No exception is thrown.	
6	**Envelope Handler availability with EVENT_FORMATTED_CB** 1-Envelope cell broadcast formatted is sent to the SIM 2-EnvelopeHandler.getTheHandler() method is called by Applet1 ~~Applet1 finalizes.~~ 3- Envelope cell broadcast formatted is sent to the SIM 4-EnvelopeHandler.getTheHandler() method is called by Applet2	1- Applet1 is triggered 2-No exception is thrown Applet1 finalizes 3- Applet2 is triggered 4-No exception is thrown	
7	**Envelope Handler availability with EVENT_UNFORMATTED_CB** 1-Envelope cell broadcast unformatted is sent to the SIM 2-EnvelopeHandler.getTheHandler() method is called by Applet1 3-EnvelopeHandler.getTheHandler() method is called by Applet2	1- Applet1 is triggered 2- No exception is thrown Applet1 finalizes 3- Applet2 is triggered 4- No exception is thrown	
8	**Envelope Handler availability with**		

Id	Description	API/Framework– Expectation	APDU Expectation
	EVENT_TIMER_EXPIRATION Timer id=1 1-Envelope Timer Expiration is sent to the SIM 2-EnvelopeHandler.getTheHandler() method is called by Applet1 Timer id=2 3-Envelope Timer Expiration is sent to the SIM 4-EnvelopeHandler.getTheHandler() method is called by Applet2 ~~Applet2 finalizes.~~	1- Applet1 is triggered 2- No exception is thrown. Applet1 finalizes 3- Applet2 is triggered 4- No exception is thrown.	
9	**Envelope Handler availability with EVENT_CALL_CONTROL_BY_SIM** 1-Envelope call control by sim is sent to the SIM 2-EnvelopeHandler.getTheHandler() method is called by Applet1	1- Applet1 is triggered 2- No exception is thrown.	
10	**Envelope Handler availability with EVENT_MO_SHORT_MESSAGE_CONTROL_BY_SIM** 1-Envelope mo short message control by sim is sent to the SIM 2-EnvelopeHandler.getTheHandler() method is called by Applet1.	1- Applet1 is triggered 2- No exception is throw	
11	**Envelope Handler availability with EVENT_EVENT_DOWNLOAD_MT_CALL** 1-Envelope event download mt call is sent to the SIM 2-EnvelopeHandler.getTheHandler() method is called by Applet1 3-EnvelopeHandler.getTheHandler() method is called by Applet2	1- Applet1 is triggered 2- No exception is thrown. Applet1 finalizes 3- Applet2 is triggered 4- No exception is thrown.	
12	**Envelope Handler availability with EVENT_EVENT_DOWNLOAD_CALL_CONNECTED** 1-Envelope event download call connected is sent to the SIM 2-EnvelopeHandler.getTheHandler() method is called by Applet1 3-EnvelopeHandler.getTheHandler() method	1- Applet1 is triggered 2- No exception is thrown. Applet1 finalizes	

Id	Description	API/Framework– Expectation	APDU Expectation
	is called by Applet2	3- Applet2 is triggered 4- No exception is thrown.	

Id	Description	API/Framework– Expectation	APDU Expectation
13	**Envelope Handler availability with EVENT_EVENT_DOWNLOAD_CALL_DISCONECTTED** 1-Envelope event download call disconnected is sent to the SIM 2-EnvelopeHandler.getTheHandler() method is called by Applet1 3-EnvelopeHandler.getTheHandler() method is called by Applet2	1- Applet1 is triggered. 2- No exception is thrown. Applet1 finalizes 3- Applet2 is triggered 4- No exception is thrown.	
14	**Envelope Handler availiability with EVENT_EVENT_DOWNLOAD_LOCATION_STATUS** 1-Envelope event download location status is sent to the SIM 2-EnvelopeHandler.getTheHandler() method is called by Applet1 3-EnvelopeHandler.getTheHandler() method is called by Applet2	1- Applet1 is triggered 2- No exception is thrown. Applet1 finalizes 3- Applet2 is triggered 4- No exception is thrown.	
15	**Envelope Handler availiability 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	1- Applet1 is triggered 2- No exception is thrown Applet1 finalizes 3- Applet2 is triggered 4- No exception is thrown	
16	**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 3-EnvelopeHandler.getTheHandler() method is called by Applet2	1- Applet1 is triggered 2- No exception is thrown. Applet1 finalizes 3- Applet2 is triggered 4- No exception is thrown.	
17	**Envelope Handler availiability with EVENT_EVENT_DOWNLOAD_CARD_READER_STATUS** 1-Envelope event download card reader status is sent to the SIM	1- Applet1 is triggered	

Id	Description	API/Framework– Expectation	APDU Expectation
	2-EnvelopeHandler.getTheHandler() method is called by Applet1 3-EnvelopeHandler.getTheHandler() method is called by Applet2	2- No exception is thrown. Applet1 finalizes 3- Applet2 is triggered 4- No exception is thrown.	

Id	Description	API/Framework– Expectation	APDU Expectation
18	**Envelope Handler availiability with EVENT_EVENT_DOWNLOAD_LANGUAGE_ SELECTION** 1-Envelope event download language selection is sent to the SIM 2-EnvelopeHandler.getTheHandler() method is called by Applet1 ~~Applet1 finalizes.~~ 3-EnvelopeHandler.getTheHandler() method is called by Applet2	1- Applet1 is triggered 2-No exception is thrown. Applet1 finalizes 3- Applet2 is triggered 4- No exception is thrown.	
19	**Envelope Handler availiability with EVENT_EVENT_DOWNLOAD_BROWSER_ TERMINATION** 1-Envelope event download browser termination is sent to the SIM 2-EnvelopeHandler.getTheHandler() method is called by Applet1 ~~Applet1 finalizes.~~ 3-EnvelopeHandler.getTheHandler() method is called by Applet2	1- Applet1 is triggered 2-No exception is thrown. Applet1 finalizes 3- Applet2 is triggered 4- No exception is thrown.	
20	**Envelope Handler availaibility with EVENT_STATUS_COMMAND** 1-Status command is sent to the SIM 2-EnvelopeHandler.getTheHandler() method is called by Applet1 3-EnvelopeHandler.getTheHandler() method is called by Applet2	1- Applet1 is triggered 2- A Toolkit exception HANDLER_NOT_AVAILABLE is thrown Applet1 finalizes 3- Applet2 is triggered 4- A Toolkit exception HANDLER_NOT_AVAILABLE is thrown	
21	**Envelope Handler availiability with EVENT_ UNRECOGNIZED_ENVELOPE** 1-An unrecognized Envelope is sent to the SIM 2-EnvelopeHandler.getTheHandler() method is called by Applet1 3-EnvelopeHandler.getTheHandler() method is called by Applet2	1- Applet1 is triggered 2- No exception is thrown. Applet1 finalizes 3- Applet2 is triggered 4- No exception is thrown.	

6.3.1.3.4 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
~~CRRN2~~	~~14,15,16,17,18,19,20,21~~
CRRC1	1,20

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

6.3.6.1.2 Test Procedure

Id	Description	API/Framework Expectation	APDU Expectation
1	**Framework checks the Cryptographic checksum and deciphers the data** Applet1 is loaded and installed 1-Envelope(SMS-PP) formatted is sent to the SIM with this features: Ciphering; Cryptographic checksum; No proof of receipt; Data = 01	1- ~~The applet~~Applet1 is triggered.	
2	**Framework checks the Cryptographic checksum and deciphers the data** Applet2 is installed 1-Envelope(SMS-PP) formatted is sent to the SIM with this features: Ciphering; Cryptographic checksum; No proof of receipt; TAR of Applet 1 Data = 02 2-Envelope(SMS-PP) 03.48 formatted is sent to the SIM with this features: No ciphering; No cryptographic checksum; No proof of receipt; TAR of Applet 2 Data = 03	1- Applet1 is triggered 3- Applet2 is triggered	2- The SIM answers to the Envelope with status words 9000 The SIM answers to the Envelope with status words 9000
3	**Envelope(SMS-PP) formatted with wrong cryptographic checksum** No ciphering;	No applet is triggered	1- The SIM answers to the Envelope with status words 9000

Id	Description	API/Framework Expectation	APDU Expectation
	Wrong Cryptographic checksum; No proof of receipt; TAR of Applet 1 Data = 04		
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.	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; 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; Cryptographic checksum; No proof of receipt; TAR of Applet 4 Data = 03	1- Applet3 is triggered 3- Applet4 is triggered	2- The SIM answers to the Envelope with status words 9000 4- The SIM answers to the Envelope with status words 9000
6	**Envelope(SMS-CB) formatted with wrong cryptographic checksum** No ciphering; Wrong Cryptographic checksum; No proof of receipt; TAR of Applet 3 Data = 04	No applet is triggered	1- The SIM answers to the Envelope with status words 9000

3GPP TSG-T3 Meeting #26
Lisbon, Portugal, 11-14 February 2003

Tdoc ⌘T3-030169
Revised T3-030038

CR-Form-v7

CHANGE REQUEST

⌘ **11.13** **CR** **A007** ⌘ **rev** **-** ⌘ Current version: **8.1.0** ⌘

*For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.*

Proposed change affects: UICC apps⌘ [X] ME [ ] Radio Access Network [ ] Core Network [ ]

Title:	⌘	Upgrade of 11.13 Specification to Release 4
Source:	⌘	T3
Work item code:	⌘	TEI
Date:	⌘	13/02/2003
Category:	⌘	**F**
Release:	⌘	Rel-4

Use one of the following categories:
F *(correction)*
A *(corresponds to a correction in an earlier release)*
B *(addition of feature),*
C *(functional modification of feature)*
D *(editorial modification)*
Detailed explanations of the above categories can be found in 3GPP TR 21.900.

Use one of the following releases:
2 (GSM Phase 2)
R96 (Release 1996)
R97 (Release 1997)
R98 (Release 1998)
R99 (Release 1999)
Rel-4 (Release 4)
Rel-5 (Release 5)
Rel-6 (Release 6)

Reason for change:	⌘	Creation of Rel 4, with good references.
Summary of change:	⌘	• References in §1, 2, 3, 4, 6.1.1.1.1, 6.1.1.2.1, 6.1.1.3.1, 6.1.1.4.1, 6.1.1.10.1, 6.1.3.3.1, 6.2.1.1.1, 6.2.4.3.4, 6.2.4.4.4, 6.2.5.4.3, 6.2.5.16.3, 6.2.9.2.1, 6.2.9.3.2, 6.2.11.1.1, 6.3.6, 6.3.9.2.1, 6.3.9.3.4, C.1, C.2.14, G.2.3, G.2.4, G.2.5
Consequences if not approved:	⌘	No release 4 available.

Clauses affected:	⌘	1, 2, 3, 4, 6.1.1.1.1, 6.1.1.2.1, 6.1.1.3.1, 6.1.1.4.1, 6.1.1.10.1, 6.1.3.3.1, 6.2.1.1.1, 6.2.4.3.4, 6.2.4.4.4, 6.2.5.4.3, 6.2.5.16.3, 6.2.9.2.1, 6.2.9.3.2, 6.2.11.1.1, 6.3.6, 6.3.9.2.1, 6.3.9.3.4, C.1, C.2.14, G.2.3, G.2.4, G.2.5

		Y	N			
Other specs affected:	⌘		X	Other core specifications	⌘	
			X	Test specifications		
			X	O&M Specifications		

Other comments:	⌘	

How to create CRs using this form:
Comprehensive information and tips about how to create CRs can be found at http://www.3gpp.org/specs/CR.htm.
Below is a brief summary:

1) Fill out the above form. The symbols above marked ⌘ contain pop-up help information about the field that they are closest to.

2) Obtain the latest version for the release of the specification to which the change is proposed. Use the MS Word "revision marks" feature (also known as "track changes") when making the changes. All 3GPP specifications can be downloaded from the 3GPP server under ftp://ftp.3gpp.org/specs/ For the latest version, look for the directory name with the latest date e.g. 2001-03 contains the specifications resulting from the March 2001 TSG meetings.

3) With "track changes" disabled, paste the entire CR form (use CTRL-A to select it) into the specification just in front of the clause containing the first piece of changed text. Delete those parts of the specification which are not relevant to the change request.

1 Scope

The present document covers the minimum characteristics considered necessary in order to provide compliance to ~~GSM~~3GPP TS ~~03~~43.019 "SIM API for Java Card ™" [7].

The present document describes the technical characteristics and methods of test for testing the SIM API for Java Card (TM) [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~~GSM 11.11~~: "~~Digital cellular telecommunication system (Phase 2+);~~ Specification of the Subscriber Identity Module - Mobile Equipment (SIM - ME) interface".

[4] 3GPP TS 51.014~~GSM 11.14~~: "~~Digital cellular telecommunications system (Phase 2+);~~ Specification of the SIM application toolkit for the Subscriber Identity Module – Mobile Equipment (SIM – ME) interface".

[5] GSM 11.17: "Subscriber Identity Module" (SIM) conformance test specification".

[6] (void)

[7] ~~GSM~~3GPP TS ~~03~~43.019 ~~Rel 98~~: "~~Digital cellular telecommunications system (Phase 2+);~~ Subscriber Identity Module Application Programming Interface (SIM API); SIM API for Java Card™; Stage 2".

[8] 3GPP TS 23.048 ~~GSM 03.48~~ Rel-4~~99~~: "~~Digital cellular telecommunications system (Phase 2+);~~ Security Mechanisms for the SIM application toolkit; Stage 2"

[9] ISO/IEC 7816-3 (1997) " Identification cards - Integrated circuit(s) cards with contacts, Part 3: Electronic signals and transmission protocols".

[10] 3GPP TS 42.019: ~~GSM 02.19~~ "~~Digital cellular telecommunications system (Phase 2+, Release 98);~~ Subscriber Identity Module Application Programming Interface (SIM API); Service description; 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".

SUN Java Card Specifications can be downloaded at http://java.sun.com/products/javacard

[14] ETSI TS 101 220 "Integrated Circuit Cards (ICC); ETSI numbering system for telecommunication; Application providers (AID)".

~~ETSI TS 101 220 "Numbering System for Telecommunication IC card applications".~~

[15] ~~GSM 11.10 1~~3GPP TS 51.010-1: "Digital cellular telecommunication system (Phase 2+); Mobile Station (MS) conformance specification; Part 1: Conformance specification".

3 Definitions and abbreviations

3.1 Definitions

The definitions specified in 3GPP TS 51.010-1~~GSM 11.10 1~~ [15] clause 3.3 shall apply, unless otherwise specified in the present clause.

Applet: An Applet is an 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 ~~03~~43.019 specification.

Expected state: the 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 ~~03~~43.019 specification.

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 ~~03~~43.019 specification.

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 this specification shall be performed taking into account the services supported by the card as specified in the EF_{SST} file.

This specification contains tests that test interoperability at the API level. This specification does not currently contain tests for interoperability at the SIM API framework and at the byte code level. These are for further study.

The test defined in this specification are applicable to cards implementing TS ~~03~~43.019 [7] ~~version 7.4.0~~ unless otherwise stated.

4.3.1.1 Conformance requirements

The conformance requirements are expressed in the following way:

- Method prototype as listed in ~~GSM~~3GPP TS ~~03~~43.019 [7]specification.
- Normal execution:
  - 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)

4.5 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 ~~GSM~~3GPP TS ~~03~~43.019[7] sim.access package.

sim.test.framework.[Test Area Reference]: Java Card packages containing Test Area References for the ~~GSM~~3GPP TS ~~03~~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 ~~GSM~~3GPP TS ~~03~~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]*

4.7.3.1 Security parameters

Loading scripts shall use the following security parameters as stated in 3GPP TS 23.048 ~~GSM 03.48~~ [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

6.1.1.1.1 Conformance Requirements

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

Normal execution

CRRN1: The constants shall have the same name and value that is defined in 3GPP TS 43.019 ~~GSM 03.19~~ [7].

6.1.1.2.1 Conformance Requirements

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

```
public short select(short fid,
                    byte[] fci,
                    short fciOffset,
                    short fciLength)
             throws java.lang.NullPointerException,
                    java.lang.ArrayIndexOutOfBoundsException,
                    SIMViewException
```

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 3GPP TS 43.019 ~~GSM 03.19~~ [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 3GPP TS 51.011 ~~GSM 11.11~~ [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.

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.

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.

6.1.1.3.1 Conformance Requirements

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

```
public void select(short fid)
           throws SIMViewException
```

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 [~~0~~43.019 - §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.

Parameter errors

No requirements.

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.

6.1.1.4.1 Conformance Requirements

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

```
public short status(byte[] fci,
                    short fciOffset,
                    short fciLength)
             throws java.lang.NullPointerException,
                    java.lang.ArrayIndexOutOfBoundsException,
                    SIMViewException
```

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 ~~GSM 03.19~~ [7], §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.

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.

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.

6.1.1.10.1 Conformance Requirements

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

```
public short increase(byte[] incr,
                      short incrOffset,
                      byte[] resp,
                      short respOffset)
               throws java.lang.NullPointerException,
                      java.lang.ArrayIndexOutOfBoundsException,
                      SIMViewException
```

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.

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.

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 (3GPP TS 51.011 ~~GSM 11.11~~: 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.

6.1.3.3.1 Conformance Requirement:

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

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]~~GSM03.19~~

CRRN2: Constructs SIMViewException a Exception with the specified reason

Parameters error

No requirements

Context errors

No requirements

6.2.1.1.1 Conformance Requirement:

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

Normal execution

CRRN1: The Toolkit Constants shall all have the same name and value defined in the 3GPP TS 43.019 [7]~~GSM03.19 normalization~~.

Parameters error

No requirements

Context errors

No requirements

6.2.4.3.4 Test Coverage

This method has only been tested with call control and the tests shall be improved during 3GPP TS 23.048 [8]~~03.48~~ tests.

CRR number	Test case number
N1	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13
N2	13
N3	6, 7
N4	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 25
N5	14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26
N6	27, 28, 29, 30, 31, 32, 33
N7	34
N8	35
C1	36
C2	37

6.2.4.4.4 Test Coverage

This method has only been tested with call control and the tests shall be improved during 3GPP TS 23.048 [8]~~03.48~~ tests.

CRR number	Test case number
N1	1, 2, 3, 4, 5, 6, 19
N2	6, 12
N3	1, 2, 3, 4, 5, 6, 19
N4	7, 8, 9, 10, 11
N5	14,15
N6	13
N7	16
C1	17
C2	18

6.2.5.4.3 Test procedure

Id	Description	API Expectation	APDU Expectation
1	**Clear the handler** `getLength()`	Result of getLength() is 0	
2	**appendTLV with length of 7** `getLength()`	Result of getLength() is 9	
3	**Clear the handler and appendTLV with Length of 250** `getLength()`	Result of getLength() is 253	
4	**Build a 7Fh Envelope response handler** `getLength()`	Result of getLength() is 81h	
5	**Build a 80h Envelope response handler** `getLength()`	Result of getLength() is 83h	

Note : Test case 3 is limited to 253 and not 256 because the current 3GPP TS 43.019 [7]~~03.19 [7]~~ is not clear enough on this point. So this test allows the two possible implementations.

6.2.5.16.3 Test procedure

Id	Description	API Expectation	APDU Expectation
1	**Call appendArray()** `length = 253`		
	Handler Overflow: Call twice the appendTLV()method	ToolkitException.HANDLER_OVERFLOW is thrown by one of the 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 `compareBuffer = 84 01 00`	Result is 00h	

4	**Successful call**
	`tag = 01h`
	`value = FEh`		

	Call copy() method		
	Compare handler `compareBuffer = 84 01 00 01 01 FE`	Result is 00h	

Note : Test case 1 call twice appendTLV because the current 3GPP TS 43.019 [7]~~03.19 [7]~~ is not clear enough on this point. So this test allows the two possible implementations.

6.2.9.2.1 Conformance requirement:

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

```
public void changeMenuEntry(byte id,
```

```
                byte[] menuEntry,
                short offset,
                short length,
                byte nextAction,
                boolean helpSupported,
                byte iconQualifier,
                short iconIdentifier)
        throws  java.lang.NullPointerException,
                java.lang.ArrayIndexOutOfBoundsException,
                ToolkitException
```

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~~GSM 11.14~~ [4].

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

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.

6.2.9.3.2 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]~~GSM 03.19 [7]~~ excepted those that aren't 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

6.2.11.1.1 Conformance requirement:

There is no API, only constants.

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]~~GSM03.19~~ .

Parameters error

No requirements

Context errors

No requirements

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]~~03.48~~ security is required in the test cases developed in the current section.

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 section 4.7.3.1 apply.

6.3.9.2.1 Conformance Requirements

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]~~GSM 03.19 []~~ - §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]~~GSM 03.19 []~~ - §5.2.).

6.3.9.3.4 Test Coverage

CRR 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 behavior in 3GPP TS 43.019 [7]~~03.19~~, which is foreseen to be corrected in future releases.

C.1 General Default Prepersonalisation

This table shows the default prepersonalisation, the file system and the files' content, that the test SIM cards shall contain unless otherwise stated.

Name	Identifier	Default Value	Special Features
EF_{ICCID}	2FE2	0F FF FF FF FF FF FF FF FF FF	This value is not compliant with 3GPP TS 51.011 ~~GSM 11.11~~[3]
EF_{IMSI}	6F07	FF FF FF FF FF FF FF FF FF	This value is not compliant with 3GPP TS 51.011 ~~GSM 11.11~~ [3]
EF_{LP}	6F05	01 FF FF FF	
EF_{Kc}	6F20	FF FF FF FF FF FF FF FF 07	
$EF_{PLMNsel}$	6F30	FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF 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 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF	
EF_{ACC}	6F78	00 00	
EF_{FPLMN}	6F7B	FF FF FF FF FF FF FF FF FF FF FF FF	
EF_{LOCI}	6F7E	FF FF FF FF 00 F0 00 00 00 FF 01	
EF_{AD}	6FAD	00 FF FF	
EF_{Phase}	6FAE	03	
EF_{FDN}	6F3B	Default value in all the records: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF	Records: 5
EF_{SMSP}	6F42	FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF	Records: 1
EF_{LND}	6F44	FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF	Records: 1
EF_{SMSS}	6F43	FF FF	
EF_{SMS}	6F3C	1st record: 00 FF … FF(length 176) 2nd record:00 FF … FF(length 176) 3rd record: 00 FF … FF(length 176)	Records: 3
EF_{ADN}	6F3A	FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF	Records: 1
EF_{CCP}	6F3D	FF FF FF FF FF FF FF FF FF FF FF FF FF FF	
EF_{MSISDN}	6F40	FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF	Records: 1
EF_{SDN}	6F49	FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF	Records: 1
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 FF FF FF FF FF FF FF FF FF FF	

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.2.14 EF_{CINA} (Cyclic Increase Not Allowed)

Identifier: '6F0D'	Structure: cyclic	Mandatory
Record length: 3 bytes	Update activity: high	

Access Conditions:

READ	ALWAYS
UPDATE	ALWAYS
INCREASE	ALWAYS (see note 1)
INVALIDATE	ALWAYS
REHABILITATE	ALWAYS

Logical Record Number	Description	Default Value	M/O	Length
1	Test Data	00 00 00	M	3 bytes
2	Test Data	00 00 00	M	3 bytes

Note 1: This file will be personalised in a way such that increase is not allowed, as indicated by the FCI byte 8, bit 7 (3GPP TS 51.011 [3]~~GSM 11.11~~: FCI structure of an EF returned by the SELECT command)

G.2.3 INSTALL(load) Section

Here are the parameters to be included in the Install(Load) command (as specified in 3GPP TS 23.048 [8]~~GSM 03.48 [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 3GPP TS 23.048 [8]~~GSM 03.48 [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])~~GSM 03.48 [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
AppletSpecificParameters	Parameters specific to the applet

The applet shall be installed with install(install and make selectable) command.