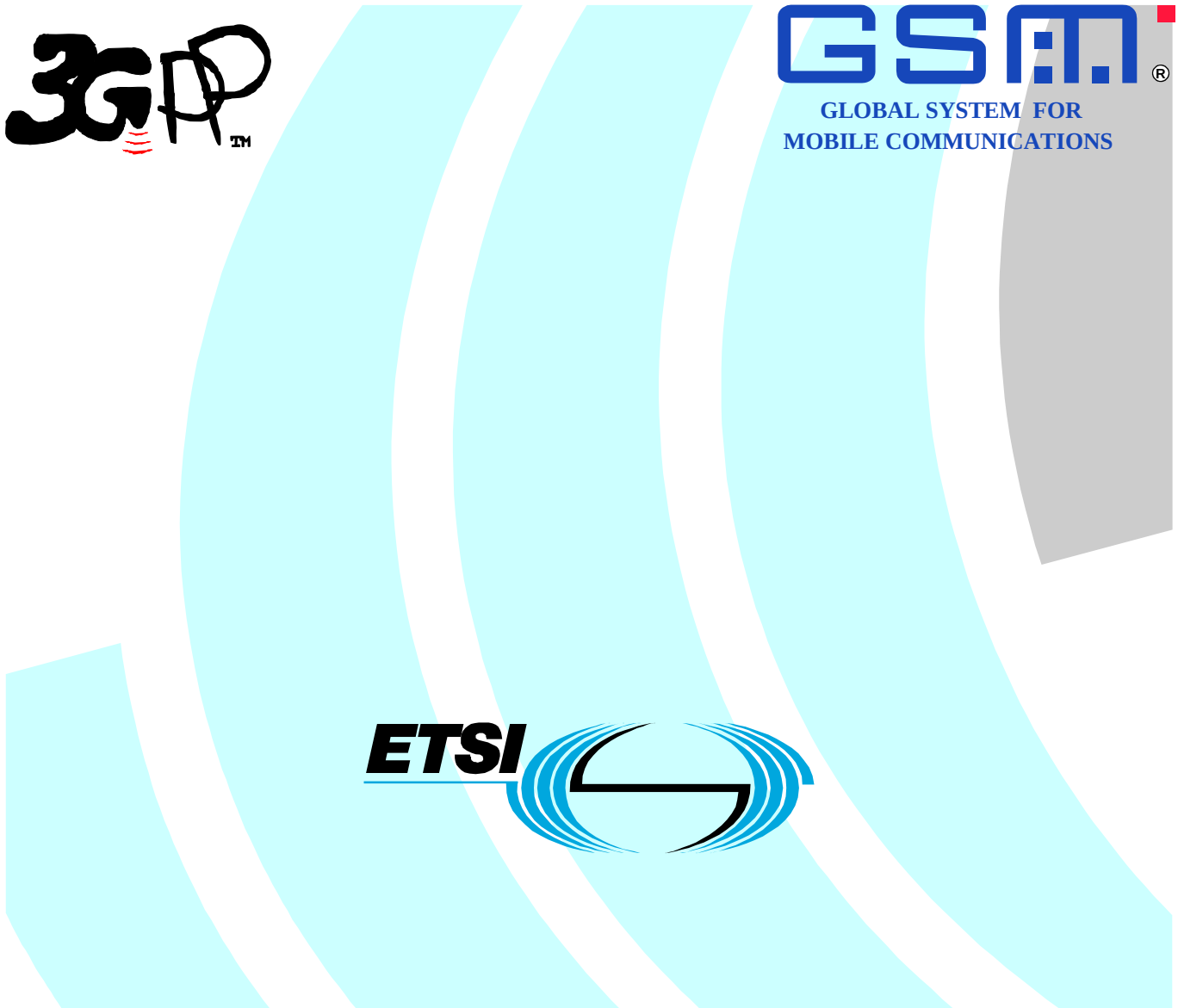
ETSI TR 131 919 V6.1.0 (2004-12)

Technical Report

**Digital cellular telecommunications system (Phase 2+);
Universal Mobile Telecommunications System (UMTS);
2G/3G Java Card(TM) Application Programming Interface (API) based applet interworking
(3GPP TR 31.919 version 6.1.0 Release 6)**



Reference
DTR/TSGT-0331919v610

Keywords
GSM, UMTS

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

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 Report (TR) 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

Foreword

This Technical Report 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.

Introduction

This document describes the different cases of interactions between SIM API and (U)SIM API.

1 Scope

The present document describes :

- Description of TS 43.019 [1] APIs and TS 102 241 [2] / TS 31.130 [7] APIs interworking.

- The behaviour and limitations of the TS 43.019 [1] APIs used in 3G environment.

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] 3GPP TS 43.019 Release 5: "Subscriber Identity Module Application Programming Interface (SIM API) for Java Card; Stage 2".

[2] ETSI TS 102 241 Release 6: "UICC Application Programming Interface (API) for Java Card™ ".

[3] 3GPP TS 51.011 Release 4: "Specification of the Subscriber Identity Module - Mobile Equipment Interface".

[4] 3GPP TS 31.102: "Characteristics of the USIM Application".

[5] 3GPP TR 31.900 Release 5: "SIM/USIM Internal and External Interworking Aspects".

[6] 3GPP TS 31.111: "USIM Application Toolkit (USAT) "

[7] 3GPP TS 31.130: "(U)SIM API for Java Card™"

[8] ETSI TS 102 221 Release 6: "Smart Cards; UICC-Terminal interface; Physical and logical characteristics".

[9] 3GPP TS 11.14 Release 99: "Specification of the SIM Application Toolkit for the Subscriber Identity Module - Mobile Equipment (SIM - ME) interface".

[10] ETSI TS 102 223 Release 6: "Smart cards; Card Application Toolkit (CAT)".

3 Definitions

For the purpose of the present document, the terms and definitions given in TS 43.019 [1], TS 102 241 [2] and the following apply.

SAT applet: applet developed using TS 43.019 [1].

(U)SAT applet: applet developed using TS 102 241 [2] and TS 31.130 [7].

SIM API: API defined in TS 43.019 [1].

(U)SIM API: API defined in TS 102 241 [2] and TS 31.130 [7].

4 The SIM API-(U)SIM API interworking

This chapter is dedicated to the interworking of the SIM API and (U)SIM API and does not consider at all the current NAA nor the current network.

It is strongly recommended that all the new applets are developed by using only the (U)SIM API. Thus, an applet loaded onto the card should use either SIM API or (U)SIM API, not both of them.

The behaviour of an applet using both APIs simultaneously is out of scope of this document.

4.1 Terminal Profile

The *MEProfile* object and *TerminalProfile* object are filled upon reception of a TERMINAL PROFILE APDU defined in TS 31.102 [4], TS 51.011 [3], TS 102 221 [8]. Their contents are identical.

A SAT applet has no issue regarding the different coding of profile data as long as TS 31.111 [6] definition is fully backward compatible with TS 11.14 [9] (i.e. there is no bit/byte swapping).

A (U)SAT applet has to take care of the bit verified in the *TerminalProfile* object because when inserted in a 2G terminal, some bits have more specific description in TS 11.14 [9] than in TS 31.111 [6]. For example, the bit 4 of the 8th byte is *'Binary choice in GET INKEY'* in TS 11.14 [9] and *'Bit = 1 if GET INKEY is supported'* in TS 31.111 [6]. For some specific features, in order to develop a (U)SAT applet independently of the terminal, it is thus recommended to interpret the bits as defined in TS 11.14 [9].

4.2 Registration and resource allocation

The terminal resources and the card resources are shared between SIM API and (U)SIM API. As a consequence, the system proactive commands generated by the card are independent of the used API. The only exception is for the alpha identifier and icon identifier of the SET UP MENU proactive command, which can differ since there are two EF_{SUME} files (one in DF_{GSM} for the SIM and one in $DF_{TELECOM}$ for USIM application). It is possible to map these two files to have the same SET UP MENU proactive command, if not mapped they may differ.

Examples:

- If an applet is registered to Call Control with (U)SIM API, an applet using SIM API can not register to Call Control.

- A timer allocated with SIM API can not be allocated by (U)SIM API.

- A menu entry identifier allocated by (U)SIM API can not be allocated by SIM API.

4.3 Triggering

No interworking issue found at the moment.

The SAT applets are triggered on their *sim.toolkit.ToolkitInterface* and the (U)SAT applets on their *uicc.toolkit.ToolkitInterface* as defined in the corresponding specifications. The order of triggering shall follow the priority level of each applet defined at its loading, independently if the applet is a SAT or a (U)SAT applet.

4.4 System handlers

The system handlers' availability for SAT applets is as defined in TS 43.019 [1] e.g. the *ProactiveHandler* may not be available if a proactive command is pending.

The system handlers' availability for (U)SAT applets is as defined in TS 102 241 [2] and TS 31.130 [7] e.g. the *ProactiveHandler* may not be available if a proactive command is pending.

As a consequence of the *EnvelopeResponseHandler* availability rules, the *EnvelopeResponseHandler* is available for all triggered SAT or (U)SAT applets, until an applet has posted an envelope response or sent a proactive command using SIM or (U)SIM API.

The content of the *sim.toolkit.EnvelopeHandler* and the content of the *uicc.toolkit.EnvelopeHandler* are identical for all triggered applets except in the case of update of EF_{SMS} where they may differ.

4.5 File access

There is no interaction between the *sim.access* package and the *uicc.access* package.

5 The behaviour and limitations of SIM API used in 3G mode

The SIM API has been designed only with the SIM as current NAA connected to a 2G network and allows application programmers access to the functions and data described in TS 51.011 [3] and TS 11.14 [9], such that SIM based services can be developed and loaded onto SIM. The SIM is mandatory in TS 43.019 [1].

This chapter points out the technical issues related to the execution of existing SAT Applets when a USIM is the current NAA or when there is no application currently selected. All these technical issues have been solved in TS 102 241 [2] and TS 31.130 [7].

5.1 File system access

The sim.access package provides a way to get a single view of the GSM (or SIM) File system defined in TS 51.011 [3]. It offers an interface (*sim.access.SIMView*) to perform operations on this file system as defined in TS 51.011 [3].

The *SIMView* object behaviour is defined in TS 43.019 [1] e.g.:

- at the invocation of the *processToolkit* method, the current file is the MF,
- the format of the File Control Information is the one defined in TS 51.011 [3],
- the access conditions as defined in TS 51.011 [3] are used.
- selection of a cyclic file is as defined in TS 51.011 [3], the record pointer shall address the record updated or increased last.

The access to any ADF is not possible, even though a USIM is the current NAA.

When two files are mapped as described in TR 31.900 [5], they have the same content after an operation (read, seek, update and increase) made by a SAT Applet.

5.2 SIM Toolkit Framework

5.2.1 Applet Triggering

In order to trigger a SAT Applet when a USIM is the current NAA or when there is no application currently selected, the SIM Toolkit Framework should be upgraded to generate events based on APDUs defined in TS 102 221 [8] and TS 31.102 [4]. As examples:

- ENVELOPE(MENU SELECTION) as defined in TS 102 223 [10] with class byte 0x80 should trigger SAT Applets registered to *EVENT_MENU_SELECTION*.
- STATUS APDU as defined in TS 102 221 [8] and in TS 31.102 [4] with class byte 0x81 should trigger SAT Applets registered to *EVENT_STATUS_COMMAND*.

- UPDATE RECORD of EF_{SMS} (EF_{SMS} in ADF USIM) APDU as defined in TS 102 221 [8] and in TS 31.102 [4] with class byte 0x02 should trigger SAT Applets registered to *EVENT_FORMATTED_SMS_PP_UPD or EVENT_UNFORMATTED_SMS_PP_UPD*.

The definition of the event *EVENT_FIRST_COMMAND_AFTER_SELECT* from TS 43.019 [1] should be replaced with:

the event *EVENT_FIRST_COMMAND_AFTER_SELECT* should only be generated by the SIM Toolkit framework when the first command is received after the ATR and before the Status Word of the processed command has been sent back.

New events/features introduced in UICC/(U)SAT API defined in TS 102 241 [2] and TS 31.130 [7] are not available for SAT applets e.g. *EVENT_DOWNLOAD_DISPLAY_PARAMETER_CHANGE*, *EVENT_DOWNLOAD_ACCESS_TECHNOLOGY_CHANGED*.

The parameters passed to the *getShareableInterfaceObject*() method to get a reference to the *ToolkitInterface* are expected to be the ones defined in TS 43.019 [1] even if a USIM is the current NAA or if there is no application currently selected. The *clientAID* parameter of this method is not specified in TS 43.019 [1].

The 3G Cell Broadcast data download is defined for the Terminal/USIM interface starting at release 5, thus this type of service would not be available for the SAT applet in earlier releases.

5.2.2 Proactive commands and responses

As long as there is compatibility between TS 11.14 [9] and TS 31.111 [6], there is no issue on proactive commands/responses issued/received by a SAT applet. New features introduced in TS 31.111 [6] are not available for SAT Applets (e.g. new REFRESH modes).

The Location Information TLV is extended with an Extended Cell identity Value TLV in TS 31.111 [6]. Thus, an existing SAT Applet may retrieve from the Cell Identity Value an information that is not complete. This TLV is present in response of a Proactive command PROVIDE LOCAL INFORMATION, in the ENVELOPE (CALL CONTROL), ENVELOPE (MO SHORT MESSAGE CONTROL), ENVELOPE (EVENT DOWNLOAD - Location status)

5.2.3 *post()* methods

The *statusType* parameter values of the *post()* methods is defined for SIM only, i.e. '9F' or '9E' (T=0 transport protocol Status Word).

If a USIM is the current NAA or if there is no application currently selected, the framework should be extended to convert the '9F' *statusType* parameter value into '61' for T=0 protocol or '9000' Status Word for T=1 protocol (Use for CALL_CONTROL_BY_SIM response, SMS-PP RP-ACK, …)

If a USIM is the current NAA or if there is no application currently selected, the framework should be extended to convert the '9E' *statusType* parameter value into '6200' warning Status Word (Use for SMS-PP RP-ERROR)

5.2.4 System proactive commands

The system proactive commands generated by the Toolkit Framework are independent of the current NAA. The only exception is for the alpha identifier and icon identifier of the SET UP MENU proactive command, which can differ since there are two EF_{SUME} files (one in DF_{GSM} for the SIM and one in $DF_{TELECOM}$ for USIM application). It is possible to map these two files to have the same SET UP MENU proactive command, if not mapped they may differ.

5.2.5 Select AID and invocation of *ProactiveHandler.send()* method

TS 43.019 [1] states that the *send*() method will never return if the GSM Applet (SIM) is deselected and another Applet is explicitly selected. When a USIM is the current NAA or when there is no application currently selected, the *send()* method should always return.

6 The behaviour and limitations of (U)SIM API used in 2G mode

No problem as the (U)SIM API are designed for this.

Annex A:
Change history

Change history							
Date	**TSG #**	**TSG Doc.**	**CR**	**Rev**	**Subject/Comment**	**Old**	**New**
2004-03	T#23	TP-040031	-		TR approved	1.3.2	6.0.0
2004-12	T#26	TP-040267	001		Correction of non-specific references to SCP documents	6.0.0	6.1.0

History

Document history		
V6.1.0	December 2004	Publication