ETSI TS 102 705 V12.0.0 (2019-05)

Smart Cards;
UICC Application Programming Interface for Java Card™
for Contactless Applications
(Release 12)

iTeh STANDARD PREVIEW
(standards.iteh.ai)
Full standards.iteh.ai/catalog/standards/sist/047217e3-20c0-42bd-9f1f-1e0717de7c0a/etsi-ts-102-705-v12.0.0-2019-05
https://standards.iteh.ai/catalog/standards/sist/047217e3-20c0-42bd-9f1f-1e0717de7c0a/etsi-ts-102-705-v12.0.0-2019-05

Reference
RTS/SCP-THCIAPIvc00

Keywords
API, smart card

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

iTeh STANDARD PREVIEW
(standards.iteh.ai)

Important notice

The present document can be downloaded from:
http://www.etsi.org/standards-search

The present document may be made available in electronic versions and/or in print. The content of any electronic and/or print versions of the present document shall not be modified without the prior written authorization of ETSI. In case of any existing or perceived difference in contents between such versions and/or in print, the prevailing version of an ETSI deliverable is the one made publicly available in PDF format at www.etsi.org/deliver.

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 https://portal.etsi.org/TB/ETSIDeliverableStatus.aspx

If you find errors in the present document, please send your comment to one of the following services:
https://portal.etsi.org/People/CommiteeSupportStaff.aspx

Copyright Notification

No part may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm except as authorized by written permission of ETSI.
The content of the PDF version shall not be modified without the written authorization of ETSI.
The copyright and the foregoing restriction extend to reproduction in all media.

© ETSI 2019.
All rights reserved.

DECT™, **PLUGTESTS**™, **UMTS**™ and the ETSI logo are trademarks of ETSI registered for the benefit of its Members.
3GPP™ and **LTE**™ are trademarks of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners.
oneM2M™ logo is a trademark of ETSI registered for the benefit of its Members and of the oneM2M Partners.
GSM® and the GSM logo are trademarks registered and owned by the GSM Association.

Contents

iTeh STANDARD PREVIEW
(standards.iteh.ai)
Full standard:
https://standards.iteh.ai/catalog/standards/sist/947217e3-20c0-42bd-9ff1-1e071de7c0a7/etsi-ts-102-705-v12.0.0-2019-05

Intellectual Property Rights

Essential patents

IPRs essential or potentially essential to normative deliverables 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 (https://ipr.etsi.org/).

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.

Trademarks

The present document may include trademarks and/or tradenames which are asserted and/or registered by their owners. ETSI claims no ownership of these except for any which are indicated as being the property of ETSI, and conveys no right to use or reproduce any trademark and/or tradename. Mention of those trademarks in the present document does not constitute an endorsement by ETSI of products, services or organizations associated with those trademarks.

Foreword

This Technical Specification (TS) has been produced by ETSI Technical Committee Smart Card Platform (SCP).

The contents of the present document are subject to continuing work within TC SCP and may change following formal TC SCP approval. If TC SCP modifies the contents of the present document, it will then be republished by ETSI with an identifying change of release date and an increase in version number as follows:

Version x.y.z

where:

- x the first digit:
  - 0 early working draft;
  - 1 presented to TC SCP for information;
  - 2 presented to TC SCP for approval;
  - 3 or greater indicates TC SCP 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.

iTeh STANDARD PREVIEW
(standards.iteh.ai)
Full standard:
https://standards.iteh.ai/catalog/standards/sist/9f7217e3-20c0-42bd-9f1f-1e0717de7c0a/etsi-ts-102-705-v12.0.0-2019-05

Modal verbs terminology

In the present document "**shall**", "**shall not**", "**should**", "**should not**", "**may**", "**need not**", "**will**", "**will not**", "**can**" and "**cannot**" are to be interpreted as described in clause 3.2 of the ETSI Drafting Rules (Verbal forms for the expression of provisions).

"**must**" and "**must not**" are **NOT** allowed in ETSI deliverables except when used in direct citation.

1 Scope

The present document describes the UICC Application Programming Interface for Java Card™ for contactless Applications. Its purpose is to provide access for a contactless Applet to the services provided by the HCI protocol defined in [4] for the communication via the CLF. In the scope of the present document contactless means support for the RF Technologies referenced by the HCI specification [4]. Low level functionality to manage gates and pipes as defined in the HCI specification [4] is not in the scope of the present document. Registration of contactless parameters and management of contactless Applets in card emulation mode is defined in "GlobalPlatform Card Specification Amendment C" [8]. Related APIs are provided in "Java Card API and Export File for Card Specification v2.2.1 (org.globalplatform) v1.5" [12] and "Java Card Contactless API and Export File for Card Specification v2.2.1 (org.globalplatform.contactless)" [13].

2 References

2.1 Normative references

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the referenced document (including any amendments) applies.

- In the case of a reference to a TC SCP document, a non-specific reference implicitly refers to the latest version of that document in the same Release as the present document.

Referenced documents which are not found to be publicly available in the expected location might be found at http://docbox.etsi.org/Reference.

NOTE: While any hyperlinks included in this clause were valid at the time of publication ETSI cannot guarantee their long term validity.

The following referenced documents are necessary for the application of the present document.

[1] ISO/IEC 7816-3 (2006): "Identification cards - Integrated circuit cards - Part 3: Cards with contacts - Electrical interface and transmission protocols".

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

[3] ETSI TS 101 220: "Smart Cards; ETSI numbering system for telecommunication application providers".

[4] ETSI TS 102 622: "Smart Cards; UICC - Contactless Front-end (CLF) Interface; Host Controller Interface (HCI)".

[5] ETSI TS 102 241: "Smart Cards; UICC Application Programming Interface (UICC API) for Java Card™".

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

[7] ETSI TS 102 226: "Smart Cards; Remote APDU structure for UICC based applications".

[8] GlobalPlatform: "GlobalPlatform Card Specification Version 2.2, Amendment C: Contactless Services" Version 1.1.

NOTE: See http://www.globalplatform.org/.

[9] ORACLE: "Application Programming Interface, Java Card™ Platform, 3.0.1 Classic Edition".

[10] ORACLE: "Runtime Environment Specification, Java Card™ Platform, 3.0.1 Classic Edition".

iTeh STANDARD PREVIEW (standards.iteh.ai) Full standard: https://standards.iteh.ai/catalog/standards/sist/947217e5-20c0-42bd-9f1f-1e0f11fe7c0a/etsi-ts-102-705-v12.0.0-2019-05

[11] ORACLE: "Virtual Machine Specification Java Card™ Platform, 3.0.1 Classic Edition".

NOTE: ORACLE Java Card Specifications can be downloaded at https://docs.oracle.com/en/java/javacard/3.1/index.html.

[12] GlobalPlatform: "Java Card API and Export File for Card Specification v2.2.1 (org.globalplatform)" v1.5.

[13] GlobalPlatform: "Java Card Contactless API and Export File for Card Specification v2.2.1 (org.globalplatform.contactless)" v1.1.

[14] ETSI TS 102 613: "Smart Cards; UICC - Contactless Front-end (CLF) Interface, Physical and data link layer characteristics".

[15] ETSI TS 102 705: "Smart Cards; UICC Application Programming Interface for Java Card™ for Contactless Applications".

2.2 Informative references

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the referenced document (including any amendments) applies.

- In the case of a reference to a TC SCP document, a non-specific reference implicitly refers to the latest version of that document in the same Release as the present document.

NOTE: While any hyperlinks included in this clause were valid at the time of publication ETSI cannot guarantee their long term validity.

The following referenced documents are not necessary for the application of the present document but they assist the user with regard to a particular subject area.

Not applicable.

iTeh STANDARD PREVIEW (standards.iteh.ai) Full standard: https://standards.iteh.ai/catalog/standards/sist/94f217e3-2006-42bd-9f1f-1e0b7de7c0a/etsi-ts-102-705-v12.0.0-2019-05

3 Definition of terms, symbols and abbreviations

3.1 Terms

For the purposes of the present document, the following terms apply:

contactless mode: used as a generic term for "Card Emulation Mode" and "Reader Mode"

contactless state: corresponds to the logical state of the contactless framework

HCP message: Message as specified in ETSI TS 102 622 [4].

NOTE: An HCP message can be of type "command", "event" or "response to a command".

RF Technology: radio frequency technology supported by the HCI (ETSI TS 102 622 [4]) protocol specification

3.2 Symbols

Void.

3.3 Abbreviations

For the purposes of the present document, the following abbreviations apply:

APDU Application Protocol Data Unit

NOTE: According to ISO/IEC 7816-3 [1].

API Application Programming Interface
CLF ContactLess Front-end

NOTE: According to ETSI TS 102 622 [4].

CRS Contactless Registry Service
HCI Host Controller Interface

NOTE: According to ETSI TS 102 622 [4].

HCP Host Controller Protocol

NOTE: According to ETSI TS 102 622 [4].

RF Radio Frequency
SCP Smart Card Platform
SWP Single Wire Protocol

NOTE: According to ETSI TS 102 613 [14].

TC Technical Committee
TS Technical Specification

4 Description

4.1 Architecture

iTeh STANDARD PREVIEW
(standards.iteh.ai)
Full standard:
https://standards.iteh.ai/catalog/standards/sist/947217e3-20c0-42bd-bf11-1e0717de7c0a/etsi-ts-102-705-v12.0.0-2019-05

The present document describes an API and a Contactless Framework that enables Java Card^{TM} Platform based Applets, defined in [9], [10] and [11], to send and receive messages using the HCI protocol as specified in ETSI TS 102 622 [4] and to act as contactless Applets. The Contactless Framework shall support card emulation mode and reader mode as specified in the HCI protocol (ETSI TS 102 622 [4]) specification.

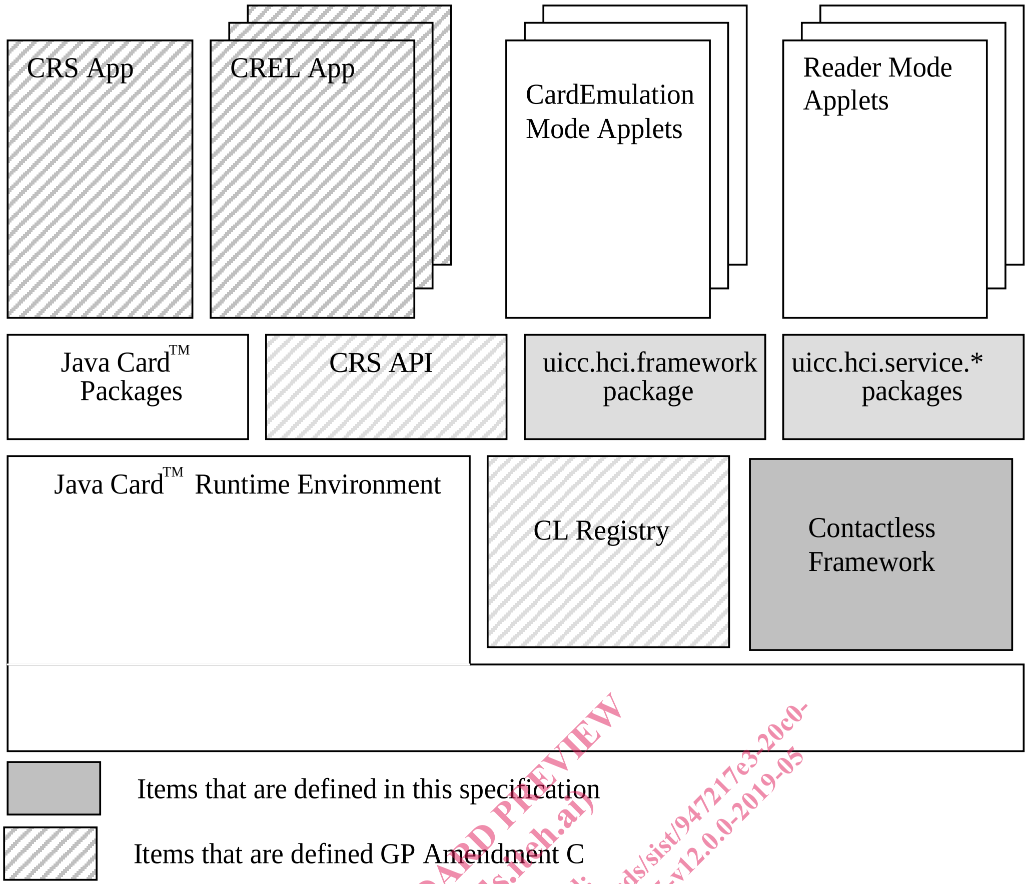

Figure 1

iTeh STANDARD PREVIEW (standards.iteh.ai) Full standard: https://standards.iteh.ai/catalog/standards/sist/947217e3-20c0-42bd-9f1f-a0f7d7eb/etsi-ts-102-705-v12.0.0-2019-05

The functionality of the Contactless Framework and the configuration of contactless parameters and the management of contactless Applets in card emulation mode are based on the functionality provided by the Contactless Registry Service (CRS), the related APIs, the CRS Application and other features and concepts which are defined in the "GlobalPlatform Amendment C" [8] and the related APIs "Java Card API and Export File for Card Specification v2.2.1 (org.globalplatform)" [12] and "Java Card Contactless API and Export File for Card Specification v2.2.1 (org.globalplatform.contactless)" [13].

The API is event driven and based on the Observer/Listener pattern. Every HCI service is encapsulated by a dedicated Service interface. These Service interfaces shall allow the registration of Listener Interfaces and the activation of events. The Listener Interfaces shall be implemented by Java objects to receive HCI messages and events in the *onCallback* method. The Registration of Listener Interfaces and activation of events shall be persistent.

An *HCIMessage* object shall encapsulates one HCP message according to the HCI protocol as specified in ETSI TS 102 622 [4]. HCI message for the different contactless modes shall be identified by different types of interfaces. It is not guaranteed that any Applet originated HCI messages are sent before the completion of the execution of the current Applet. The Contactless Framework sends the Applet originated HCI messages in the same order as they are submitted by the Applet.

NOTE 1: The Contactless Framework may not have enough resources to send several HCI messages submitted during the same *onCallback* method execution. The Applet should be aware of this limitation (e.g. use suitable error handling strategy, or send only one HCI message in the *onCallback* method at a time).

Any *onCallback()* method of a Listener interface shall not be invoked again while another *onCallback()* method is still being executed. The Contactless Framework shall be able to receive one or more HCI messages while waiting for a response related to a command originated by the Applet (e.g. processing a request for parameters) especially for the EVT_FIELD_OFF case.

The HCI event EVT_FIELD_OFF shall be buffered and sent by the Contactless Framework as soon as the Contactless Framework becomes the current context.

All other HCI messages shall be delivered to the Applet instance in the same order as they were received by the Contactless Framework.

Contactless State is the logical state of the Contactless Framework it can take the value enabled and disabled. It refers to the "contactless functionality in the UICC" as used in ETSI TS 102 223 [6]

This state can be changed with the mechanisms defined in ETSI TS 102 223 [6], and by the method *setCommunicationInterface()* API method of "GlobalPlatform Amendment C" [8].

The Contactless State applies only to the Card Emulation Mode and the Reader Mode, and it does not apply to the Connectivity service.

When the Contactless State is disabled, the Contactless Framework shall throw an *HCIException* with reason code HCI_CURRENTLY_DISABLED when an Applet invokes a method which requires that the Contactless State is enabled.

When the Contactless State is enabled and the state of the SWP [14] interface is DEACTIVATED and when the Contactless Framework needs to send data over the SWP [14] interface then it shall send the proactive command ACTIVATE defined in ETSI TS 102 223 [6] if supported by the terminal.

NOTE 2: An Applet may use the method *HCIDevice.isHCIServiceAvailable()* to check if the Contactless Framework supports sending the ACTIVATE command on pre Rel-11 implementations.

The underlying HCI communication layer as defined in ETSI TS 102 622 [4] provides reliable message transfer. Therefore no errors can be reported to the application layer. For this reason no error reporting and recovery mechanism related to HCI communication are defined in the present document.

The API is split into two parts. One is a generic framework that provides a factory class to retrieve the different Service instances that are provided by the HCI implementation, and that allows discovery of whether the UICC is inserted into a HCI network. The second part of the API implements the Services that are defined for the HCI protocol, card emulation mode, reader mode and connectivity service. The support of the package implementing reader mode, *uicc.hci.services.readermode*, is optional.

iTeh STANDARD PREVIEW (standards.iteh.ai) Full standard: https://standards.iteh.ai/catalog/standards/sist/04f17e3-20c6-42b0-ff-1e7f1de7c0a/etsi-ts-102-705-v12.0.0-2019-05

4.2 Card Emulation Mode

In card emulation mode there exist two exclusive ways to exchange messages over the HCP [4]. The first is based on APDUs provided to the Applet through its *process()* method as specified in "Application Programming Interface, Java Card™ Platform, 3.0.1 Classic Edition" [9]. The second is made available by the package *uicc.hci.services.cardemulation* defined in the present document.

The *uicc.hci.services.cardemulation* package shall provide the communication technologies for the card emulation mode defined by the HCP as specified in [4]. The Contactless Framework shall bind the services defined in the *uicc.hci.services.cardemulation* package to the underlying HCI resources (e.g. gates and pipes) defined in the HCI architecture as specified in [4]. The parameters to be used by the HCI layer may be provided to the framework as defined in "GlobalPlatform Amendment C" [8].

In case of a communication error on the RF interface (i.e. the RF error indicator is set), messages are not propagated to the application layer in CardEmulation Mode.

For the API defined in the present document the card emulation capability shall be provided to Applets through a service interface implemented by the Contactless Framework. Applet instances shall receive *CardEmulationMessages* after the registration of a *CardEmulationListener* interface to a *CardEmulationService* only if the EVENT_ON_SEND_DATA is activated for the Applet instance. If the EVENT_ON_SEND_DATA is deactivated for the Applet instance and an APDU is received via the EVT_SEND_DATA, the *javacard.framework.APDU* class and the *process()* method of the Applet instance shall be invoked.

It shall not be possible to switch between the usage of the *CardEmulationListener* interface and the invocation through the *process()* method within a contactless application session, i.e. not before the Applet has been deselected and selected again. Applets communicating through the *process()* method shall also be able to use the API services defined in the present document which do not require a *CardEmulationListener* registration (e.g. requesting the power mode or connectivity service).