3GPP TSG-CT Meeting #28
Quebec, 1-3 June 2005

CP-050141

Agenda Item: 10.14

Source: CT6

Title: USSD CRs

Document for: Approval

This document contains the following change requests that are agreed by 3GPP TSG CT WG6 and forwarded to 3GPP TSG CT plenary for approval:

Table of USSD CRs

CT doc	CT6 Doc	Spec	CR	Rev	Rel	Title	Source	Cat	WI	Agenda	Status
CP-050141	C6-050420	31.130	014		Rel-7	Addition of new events EVENT_FORMATTED_USSD and EVENT_UNFORMATTED_USSD	CT6	F	USSD	12.3.2	Agreed
CP-050141	C6-050419	31.111	144		Rel-7	Clarification on the coding of the DCS field for USSD data download	CT6	F	USSD	12.1.1	Agreed
CP-050141	C6-050418	31.115	005		Rel-7	Introduction of secured data download for USSD	CT6	B	USSD	13.1	Agreed

3GPP TSG-CT6 #35
Cancun, Mexico, 25-29 April 2005

C6-050418

CR-Form-v7.1

CHANGE REQUEST

⌘ **31.115** **CR** 005 ⌘ **rev** **-** ⌘ Current version: **6.4.0** ⌘

*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:	⌘	Introduction of secured data download for USSD
Source:	⌘	CT6
Work item code:	⌘	USSD ***Date:*** ⌘ 28/04/2005
Category:	⌘	**B** ***Release:*** ⌘ Rel-7

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)*
Rel-7	*(Release 7)*

Reason for change:	⌘	Following the release 6 WIs from SA1 and T3 about USSD message transfer to USIM, it was intended and agreed to add secured data download to USSD.
Summary of change:	⌘	The concept of secured download is introduced for USSD. The transport related operations are described separately in a new annex.
Consequences if not approved:	⌘	

Clauses affected:	⌘	1, 2, 3.2, 4.4, 6 (new), 7 (new), Annex X (new)

		Y	N		
Other specs affected:	⌘		X	Other core specifications	⌘
			X	Test specifications	
			X	O&M Specifications	
Other comments:	⌘	New Response Status code '0C' should be reserved by ETSI SCP in ETSI TS 102 225. See LS in C6-050456.			

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 specifies the structure of the Secured Packets in implementations using Short Message Service Point to Point (SMS-PP), ~~and~~ Short Message Service Cell Broadcast (SMS-CB), and Unstructured Supplementary Service Data (USSD) based on TS 102 225 [9].

The structure of the Secured Packets shall comply with the one defined in TS 102 225 [9]. The present document only contains additional requirements or explicit limitations for SIM/USIM applications.

It is applicable to the exchange of secured packets between an entity in a 3G or GSM PLMN and an entity in the (U)SIM.

Secured Packets contain application messages to which certain mechanisms according to TS 102 224 [2] have been applied. Application messages are commands or data exchanged between an application resident in or behind the 3G or GSM PLMN and on the (U)SIM. The Sending/Receiving Entity in the 3G or GSM PLMN and the UICC are responsible for applying the security mechanisms to the application messages and thus turning them into Secured Packets.

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 and/or edition number or version number) 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 TR 21.905: "Vocabulary for 3GPP Specifications".

[2] ETSI TS 102 224 Release 6: "Smart Cards; Security mechanisms for UICC based Applications - Functional requirements".

[3] 3GPP TS 23.040: "Technical realization of the Short Message Service (SMS)".

[4] 3GPP TS 24.011: "Point-to-Point (PP) Short Message Service (SMS) support on mobile radio interface".

[5] ISO/IEC 7816-6 (1996): "Identification cards - Integrated circuit(s) cards with contacts - Part 6: Interindustry data elements".

[6] 3GPP TS 23.041: "Technical realization of Cell Broadcast Service (CBS)".

[7] 3GPP TS 24.012: "Short Message Service Cell Broadcast (SMSCB) support on the mobile radio interface".

[8] 3GPP TS 23.038: "Alphabets and language-specific information".

[9] ETSI TS 102 225 Release 6: "Smart Cards; Secured packet structure for UICC based applications".

[xx] 3GPP TS 24.090: "Unstructured Supplementary Service Data (USSD) - Stage 3".

3.1 Definitions

For the purposes of the present document, the terms and definitions given in TS 102 225 [9] and the following apply:

Message Identifier: two-octet field used to identify the source and type of the message

Page Parameter: single octet field used to represent the CBS page number in the sequence and the total number of pages in the SMS-CB message

Serial Number: two octet field which identifies a particular message
It is linked to the Message Identifier and is altered every time the message is changed

Short Message: information that may be conveyed by means of the SMS Service as defined in 3GPP TS 23.040 [3].

USSD message: information that may be conveyed in the USSD-String field of a Facility message as defined in 3GPP TS 24.090 [xx].

3.2 Abbreviations

For the purpose of the present document, the abbreviations given in TS 102 225 [9] and the following apply:

CBC	Cipher Block Chaining
CBS	Cell Broadcast Service
CCF	Concatenation Control Field
DCS	Data Coding Scheme
IEI	Information Element Identifier
IEIDL	Information Element Identifier Data Length
IED	Information Element Data
MID	Message IDentifier
MO-SMS	Mobile Originated Short Message Service
MT-SMS	Mobile Terminated Short Message Service
PFI	Packet Format Information
PLMN	Public Land Mobile Network
PP	Page Parameter
SIM	Subscriber Identity Module
SM	Short Message
SMS	Short Message Service
SMS-PP	Short Message Service - Point to Point
SMS-CB	Short Message Service - Cell Broadcast
SMS-SC	Short Message Service - Service Centre
SN	Serial Number
UM	USSD message
USIM	Universal Subscriber Identity Module
USSD	Unstructured Supplementary Service Data

4 Implementation for SMS-PP

[…]

4.4 Structure of the Response Packet

The Response Packet is as follows. This message is generated by the Receiving Entity and possibly includes some data supplied by the Receiving Application, and returned to the Sending Entity/Sending Application. In the case where the Receiving Entity is the UICC, depending on bit 6 of the second octet of the SPI, this Response Packet is generated on the UICC, either:

- retrieved by the ME from the UICC, and included in the User-Data part of the SMS-DELIVER-REPORT returned to the network; or
- fetched by the ME from the UICC after the Send Short Message proactive command.

The structure of an SMS-DELIVER/SUBMIT User Data object is defined in TS 23.040 [3].

RPI identifies the Response Packet and indicates that the first portion of the SM (8 bit data) contains the Response Packet Length (RPL), the Response Header Length (RHL) followed by the remainder of the Response Header: the Secured Data follows on immediately as the remainder of the SM element.

The relationship between the Response Packet and its inclusion in the UDH structure of a single Short Message defined in TS 23.040 [3] is as following:

- RPI is mapped to IEIa defined in TS 23.040 [3] and shall be set to '71'.
- IEDa defined in TS 23.040 [3] shall be a null field and its length IEIDLa shall be set to '00'.

The following Table 3 indicates the Response Packet contained in a single SMS-PP. It is a particular implementation for single SMS-PP of the generic Response Packet structure described in TS 102 225 [9].

Table 3: Structure of the Response Packet contained in the SM (8 bit data)

Generalised Response Packet Elements (Refer to table 3)	Length	Description
Response Packet Length	2 octets	Length of the Response Packet (RPL), coded over 2 octets, and shall not be coded according to ISO/IEC 7816-6 [5]. (see note)
Response Header Identifier		(RHI) Null field.
Response Header Length	1 octet	Length of the Response Header (RHL), coded over one octet, and shall not be coded according to ISO/IEC 7816-6 [5].
TAR to RC/CC/DS elements in the Response Header	Variable	The remainder of the Response Header as described in TS 102 225 [9]. Response Status Codes are defined in clause 7.
Secured Data	Variable	Additional Response Data (optional), including padding octets as described in TS 102 225 [9].

NOTE: This field is not absolutely necessary but is placed here to maintain compatibility with the structure of the Command Packet when included in a SMS-SUBMIT or SMS-DELIVER.

In order to achieve a modulo 8 length of the data before the RC/CC/DS field in the Response Header, the Length of the Response Packet, the Length of the Response Header and the three preceding octets (UDHL, IEIa and IEIDLa defined in TS 23.040 [3]) shall be included in the calculation of RC/CC/DS if used. These fields shall not be ciphered.

~~**Table 4: Response Status Codes**~~

~~**Status Code (hexadecimal)**~~	~~**Meaning**~~
~~'00' to '0A'~~	~~See TS 102 225 [9]~~
~~'0B'~~	~~Actual response data to be sent using SMS-SUBMIT.~~
~~'0C' - 'FF'~~	~~See TS 102 225 [9]~~

6 Implementation for USSD

The USSD application mode enables the transparent transport of data between an application residing in the network and a UICC based application. In such a case, to secure the payload of USSD operations, security mechanisms defined in TS 102 225 [9] shall be applied to the USSD messages. Generic secured Command Packet and secured Response Packet as defined in TS 102 225 [9] are contained, as defined hereafter, in the UM part of the USSD String. The USSD String shall be formatted according to annex X, where the PFI byte indicates that Application Data are formatted according to the present document.

The Data Coding Scheme of the USSD String (as defined in TS 23.038 [8]) shall be set to 0x96 (DCS = '10010110') to indicate that data is binary (8 bit data), and formatted according to annex X. In USSD Application mode, which uses an 8-bit character set, the maximum length of the USSD String field is 160 bytes.

Command and Response packets exceeding 159 bytes shall be segmented as described in sections 6.2 and 6.4.

6.1 Structure of the Command Packet contained in a Single USSD Message

The UM field of an USSD String contains the Command Packet.

The Command Packet shall be coded as the generic Command Packet described in TS 102 225 [9].

In the Command Packet, the Command Packet Identifier (CPI) value is '03' and the Command Header Identifier (CHI) is a Null field.

CPI, CPL and CHL shall be included in the calculation of the RC/CC/DS.

The SPI shall be coded as specified in TS 102 225 [9].

6.2 Structure of the Command Packet contained in concatenated USSD Messages

If the Command Packet, which is structured as described in section 6.1, is longer than 159 bytes (including the Command Header) then it shall be handled as follows.

- The entire Command Packet including the Command Header shall be separated into its component concatenated parts.
- The Command Packet is handled as a Concatenated USSD Message as described in annex X of the present document.
- The Command Packet Header will only be present in the first segment of a concatenated message.

If the data is ciphered, then it is ciphered as described above, before being broken down into individual concatenated elements.

CPI, CPL and CHL shall be included in the calculation of the RC/CC/DS.

The SPI shall be coded as specified in TS 102 225 [9].

An example illustrating a Command Packet split over a sequence of three messages is shown below.

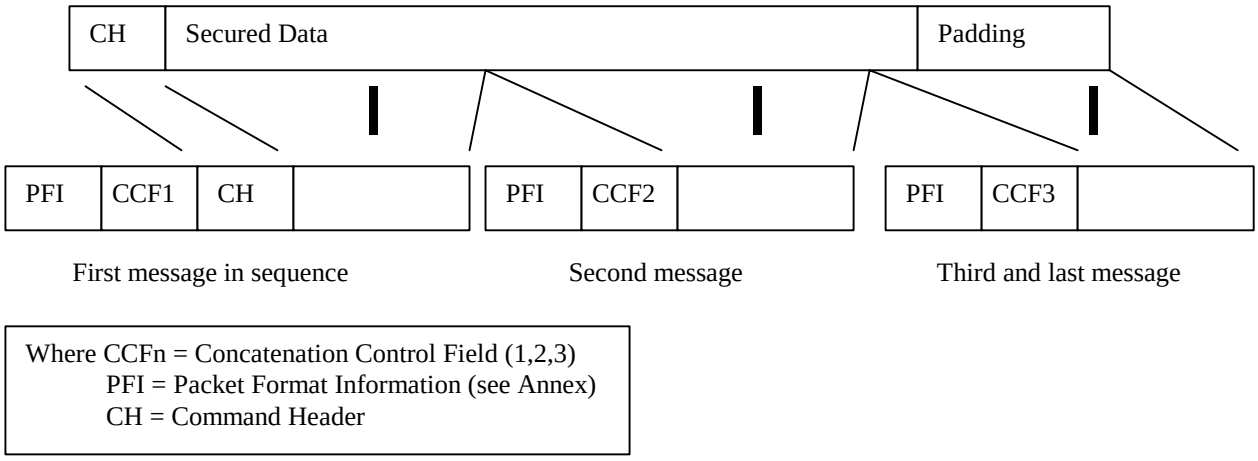

Figure 4: Example of command split using concatenated USSD messages

6.3 Structure of the Response Packet

The Response Packet is generated by the Receiving Entity and possibly includes some data supplied by the Receiving Application, and returned to the Sending Entity/Sending Application. In the case where the Receiving Entity is the UICC, this Response Packet is generated on the UICC, retrieved by the ME from the UICC, and included in the Return Result Component of a Facility message (see TS 24.090 [xx]) returned to the network.

The USSD operations are defined in TS 24.090 [xx].

The UM field of an USSD String contains the Response Packet.

The Response Packet shall be coded as the generic Response Packet described in TS 102 225 [9].

In the Response Packet, the Response Packet Identifier (RPI) value is '04' and the Response Header Identifier (RHI) is a Null field.

RPI, RPL and RHL shall be included in the calculation of the RC/CC/DS.

Coding of Response Status Codes is defined in clause 7.

6.4 Structure of the Response Packet contained in concatenated USSD Messages

If the Response Packet, which is structured as described in section 6.3, is longer than 159 bytes (including the Response Header) then it shall be handled as follows.

- The entire Response Packet including the Response Header shall be separated into its component concatenated parts.
- The Response Packet is handled as a Concatenated USSD Message as described in annex X of the present document.
- The Response Packet Header will only be present in the first segment of a concatenated message.

If the data is ciphered, then it is ciphered as described above, before being broken down into individual concatenated elements.

RPI, RPL and RHL shall be included in the calculation of the RC/CC/DS.

An example illustrating a Response Packet split over a sequence of three messages is shown below.

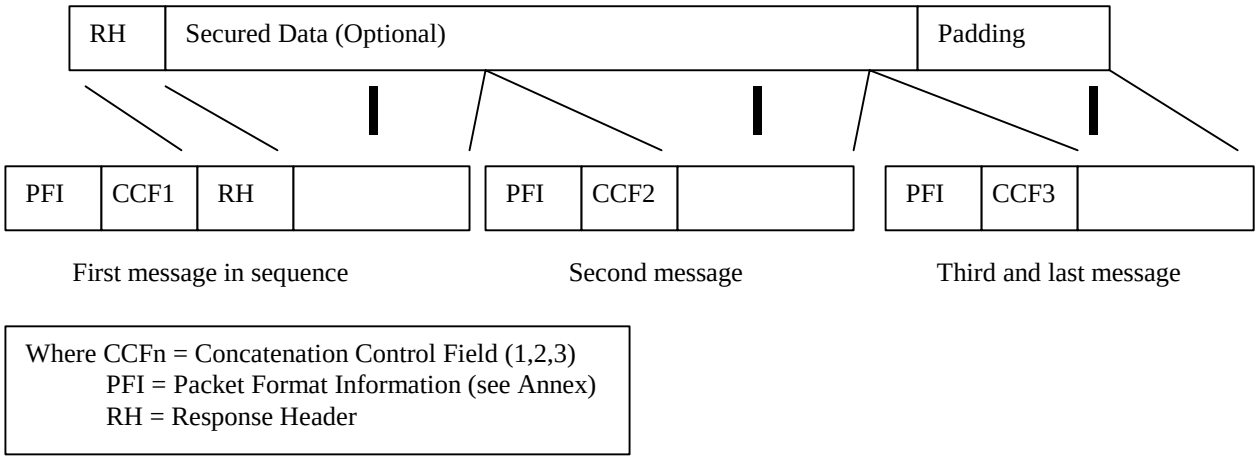

Figure 5: Example of Response split using concatenated USSD messages

If it is indicated in the SPI2 of a Command Packet to send back a PoR and if the Response Packet is too large to be contained in a single USSD String, then:

- One single Response Packet shall be sent back to the SE using the Return Result Component contained in the subsequent Facility message. This Response Packet:
  - Shall not contain any additional response data
  - Shall contain the Response Status Code set to '0C' ('Actual response data to be sent using a ProcessUnstructuredSS-Request invoke component (i.e. using SEND USSD proactive command) ').
  - The security applied to this Response Packet shall be the one indicated in the SPI2 of the Command Packet.
- This shall be followed by a complete Response Packet, contained in a concatenated USSD Message as defined above.

7. Specific Response Status Codes

Status Code (hexadecimal)	Meaning
'00' to '0A'	See TS 102 225 [9]
'0B'	Actual response data to be sent using SMS-SUBMIT. See section 4.4.
'0C'	Actual response data to be sent using a ProcessUnstructuredSS-Request invoke component (i.e. using Send USSD proactive command). See section 6.3
'0D' - 'FF'	See TS 102 225 [9]

Specific Response Status Codes

Annex X (normative): USSD String format

For the purpose of UICC-based application, the USSD String shall be coded as follows:

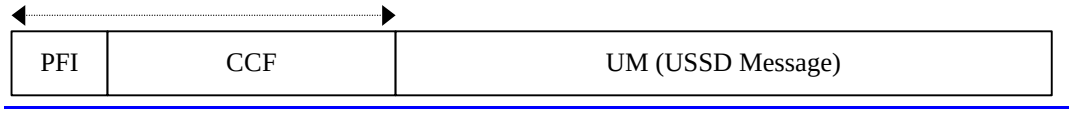

Figure 6: USSD String format

The header of an USSD Message may contain two fields:

- A mandatory PFI field, which is coded on 1 byte. The PFI contains information on the format of the USSD String.

- An optional CCF field, which is coded on 3 bytes. The CCF field presence is indicated by the PFI.

The PFI is coded as follows.

b8	b7	b6	b5	b4	b3	b2	b1	
					X	0	0	Proprietary Application Data format
					X	0	1	Application Data formatted according to the present document. If b2 b1 = '01' (Application Data formatted according to the present document), then b3 shall be coded as follows:
					0	0	1	No CCF field
					1	0	1	CCF field present
								Reserved for future use

The usage of CCF field allows USSD Messages to be concatenated to form a longer message. The CCF field contains information set by the application so that the receiving entity is able to re-assemble the received UMs in the correct order. Additionally, the CCF contains a reference number, which allows the receiving entity to discriminate between messages. The CCF octets shall be coded as follows.

Octet 1: Concatenated USSD Message reference number.

This octet shall contain a modulo-256 counter indicating the reference number for a particular USSD Message, Concatenated or not. This reference number shall remain constant for every USSD Message that makes up a particular Concatenated USSD Message.

Octet 2: Total number of USSD Messages in the Concatenated USSD Message.

This octet shall contain a value in the range 1 to 255 indicating the total number of USSD Messages constituting the Concatenated USSD Message. The value shall start at 1 and remain constant for every USSD Message that makes up the Concatenated USSD message. If the value is zero then the receiving entity shall ignore the whole USSD Message.

Octet 3: Sequence number of the current USSD Message.

This octet shall contain a value in the range 1 to 255 indicating the sequence number of a particular USSD Message within the Concatenated USSD Message. The value shall start at 1 and increment by one for every USSD Message sent within the Concatenated USSD Message. If the value is zero or the value is greater than the value in octet 2 then the receiving entity shall ignore the whole USSD Message.

The UM field contains the actual application data (e.g. secure Command/Response Packets coded according to the present document).

In each USSD String in a concatenated series, the PFI and CCF fields shall be present.

3GPP TSG-CT6 #35
Cancun, Mexico, 25-29 April 2005

C6-050419

CR-Form-v7.1

CHANGE REQUEST

⌘ **31.111** **CR** 144 ⌘ **rev** **-** ⌘ Current version: **6.5.0** ⌘

*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:	⌘	Clarification on the coding of the DCS field for USSD data download
Source:	⌘	CT6
Work item code:	⌘	USSD
Date:	⌘	28/04/2005
Category:	⌘	**B**
Release:	⌘	Rel-7

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)*
Rel-7	*(Release 7)*

Reason for change:	⌘	The USSD data download has been introduced in the release 6 of TS 31.111 without considering the definition of USIM-specific USSD data format. The definition of such a format is required to: - re-use the security packets as defined in TS 102 225. - fullfill the requirements of SA1 and T3 to introduce the secured data download via USSD.
Summary of change:	⌘	Add a note to indicate that if DCS is set to a specific value (0x96) which was not used previously for USSD Data download, the USSD string is formatted according to TS 31.115.
Consequences if not approved:	⌘	Existing security mechanisms cannot be reused in a standard manner.

Clauses affected:	⌘	2, 7.6.1, 8.17

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

Other comments:	⌘	This document is related to C6-050418

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.

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 22.002: "Circuit Bearer Services (BS) supported by a Public Land Mobile Network (PLMN)".

[2] 3GPP TS 22.030: "Man-Machine Interface (MMI) of the User Equipment (UE)".

[3] 3GPP TS 22.042: "Network Identity and Time Zone (NITZ); Service description; Stage 1".

[4] 3GPP TS 23.038: "Alphabets and language-specific information".

[5] 3GPP TS 23.040: "Technical realization of the Short Message Service (SMS)".

[6] 3GPP TS 23.041: "Technical realization of Cell Broadcast Service (CBS)".

[7] 3GPP TS 23.122: "Non-Access Stratum functions related to Mobile Station (MS) in idle mode".

[8] 3GPP TS 24.007: "Mobile radio interface signalling layer 3; General aspects".

[9] 3GPP TS 24.008: "Mobile radio interface layer 3 specification; Core network protocols; Stage 3".

[10] 3GPP TS 24.011: "Point-to-Point (PP) Short Message Service (SMS) support on mobile radio interface".

[11] 3GPP TS 24.080: "Mobile radio layer 3 supplementary services specification; Formats and coding".

[12] 3GPP TS 27.007: "AT command set for 3G User Equipment (UE)".

[13] 3GPP TS 31.101: "UICC-terminal interface; Physical and logical characteristics".

[14] 3GPP TS 31.102: "Characteristics of the USIM application".

[15] Void.

[16] Void.

[17] Void.

[18] Void.

[19] Void.

[20] Void.

[21] Void.

[22] 3GPP TS 22.001: "Principles of circuit telecommunication services supported by a Public Land Mobile Network (PLMN)".

[23] 3GPP TS 23.048: "Security Mechanisms for the (U)SIM application toolkit; Stage 2".

[24] Void.

[25] Void.

[26] Void.

[27] 3GPP TS 44.018: "Mobile radio interface Layer 3 specification; Radio Resource Control Protocol".

[28] Void.

[29] Void.

[30] 3GPP TS 23.003: "Numbering, addressing and identification".

[31] Void.

[32] ETSI TS 102 223 Release 6: "Smart Cards; Card Application Toolkit".

[33] 3GPP TR 21.905: "Vocabulary for 3GPP specifications".

[34] 3GPP TS 22.101: "Service aspects; Service principles".

[35] 3GPP TS 25.401: "UTRAN overall description".

[36] 3GPP TS 25.413: "UTRAN Iu interface RANAP signalling".

[37] 3GPP TS 24.090: "Unstructured Supplementary Service Data (USSD) - Stage 3".

[38] 3GPP TS 25.331: "Radio Resource Control (RRC) Protocol Specification".

[39] 3GPP TS 25.133: "Requirements for support of radio resource management".

[40] 3GPP TS 23.140: "Multimedia Messaging Service (MMS), Stage 2".

[xx] 3GPP TS 31.115: "Secured packet structure for the (U)SIM Toolkit applications".

7.6.1 Procedure

If the service "data download via USSD and USSD application mode" is allocated and activated in the USIM Service Table (see TS 31.102 [14]), then the ME shall follow the procedure below:

When the ME receives a USSD packet it shall pass the message transparently to the USIM using the ENVELOPE (USSD DOWNLOAD) if the Data Coding Scheme of the USSD message (as defined for the CBS Data Coding Scheme ~~in the General Data Coding Indication described for the CBS / USSD DSC~~ in TS 23.038 [4]) indicates the USIM as the target (Bit 0 set to 0 and Bit 1 set to 1):

- The ME shall wait for an acknowledgement from the USIM:

- if the UICC responds with '90 00', the ME shall acknowledge the receipt of USSD message to the network using a FACILITY message. The ME will supply the response data from the UICC in the USSD String of the return result component of the FACILITY message it will send back to the network (see TS 24.090 [37]). The alphabet and language indicators shall be those used in the original message.

- if the USIM responds with '93 00', the ME shall either retry the command or send back a FACILITY message to the network. The ME will supply the status word followed by the response data from the UICC in the USSD String of the return result component of the FACILITY message it will send back to the network (see TS 24.090 [37]). The alphabet and language indicators shall be those used in the original message.

- if the UICC responds with '62 XX' or '63 XX', the ME shall acknowledge the receipt of the USSD message to the network using a FACILITY message. The ME will supply the status word followed by the response data from the UICC in the USSD String of the return result component of the FACILITY message it will send back to the network (see TS 24.090 [37]). The alphabet and language indicators shall be those used in the original message.

If the service "data download via USSD and USSD application mode " is not allocated and activated in the USIM Service Table, and the ME receives a USSD message with a Data Coding Scheme indicating that the destination is the card (as defined above), the ME shall return a FACILITY message to the network. The ME will supply the status word '6D 00' (i.e. Instruction code not supported or invalid) in the USSD String of the return result component of the FACILITY message it will send back to the network (see TS 24.090 [37]). The alphabet and language indicators shall be those used in the original message.

8.17 USSD string

Byte(s)	Description	Length
1	USSD string tag	1
2 to (Y-1)+2	Length (X)	Y
(Y-1)+3	Data coding scheme	1
(Y-1)+4 to (Y-1)+X+2	USSD string	X-1

The Data coding scheme is coded as for Cell Broadcast defined in 3GPP TS 23.038 [4]. The coding of the USSD string is defined in 3GPP TS 22.030 [2].

NOTE1: The MMI mode uses a 7 bit character set, the Application mode uses a 8 bit character set.

NOTE2: The DCS is set to 0x96 to indicate that the USSD string is formatted according to TS 31.115 [xx].

3GPP TSG-CT6 #35 **C6-050420**
Cancun, Mexico, 25-29 April 2005

CR-Form-v7.1

CHANGE REQUEST

⌘ **31.130** **CR** 014 ⌘**rev** **-** ⌘ Current version: **7.0.0** ⌘

*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:	⌘	Addition of new events EVENT_FORMATTED_USSD and EVENT_UNFORMATTED_USSD
Source:	⌘	CT6
Work item code:	⌘	USSD
Date:	⌘	28/04/2005
Category:	⌘	**B**
Release:	⌘	Rel-7

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)*
Rel-7	*(Release 7)*

Reason for change:	⌘	Following the release 6 WIs from SA1 and T3 about USSD message transfer to USIM, it was intended and agreed to add secured data download to USSD.
Summary of change:	⌘	Addition of new events EVENT_FORMATTED_USSD and EVENT_UNFORMATTED_USSD. Support of USSD Data Download in getSecuredDataLength() and getSecuredDataOffset() methods.
Consequences if not approved:	⌘	
Clauses affected:	⌘	6.2, 6.3, 6.5, Annex A

Other specs affected: ⌘

Y	N	
	X	Other core specifications
	X	Test specifications
	X	O&M Specifications

⌘

Other comments:	⌘	Linked to CR to TS 31.115 C6-050418 New Events values should be reserved by ETSI SCP in ETSI TS 102 241 See LS in C6-050456.

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.

6.2 Definition of Events

The following events can trigger a Toolkit Applet in addition to the events defined in TS 102 241[2], all short values are reserved in TS 102 241[2]:

Table 1: (U)SAT event list

Event Name	Reserved short value
EVENT_FORMATTED_SMS_PP_ENV	2
EVENT_FORMATTED_SMS_PP_UPD	3
EVENT_UNFORMATTED_SMS_PP_ENV	4
EVENT_UNFORMATTED_SMS_PP_UPD	5
EVENT_UNFORMATTED_SMS_CB	6
EVENT_MO_SHORT_MESSAGE_CONTROL_BY_NAA	10
EVENT_FORMATTED_SMS_CB	24
EVENT_FORMATTED_USSD	121
EVENT_UNFORMATTED_USSD	122

EVENT_FORMATTED_SMS_PP_ENV, EVENT_UNFORMATTED_SMS_PP_ENV, EVENT_FORMATTED_SMS_PP_UPD, EVENT_UNFORMATTED_SMS_PP_UPD

There are two ways for a card to receive a Short Message Point to Point: via an ENVELOPE(SMS-PP DOWNLOAD) APDU as defined in TS 31.111[7] and TS 51.014[8] or an UPDATE RECORD EF_{SMS} APDU as defined in TS 31.102[3] and TS 51.011[4]. The EF_{SMS} can be either located under the $DF_{Telecom}$ or under any ADF as defined in TS 31.102[3] and TS 51.011[4].

The received Short Message may be:

- formatted according to TS 31.115[9] or an other protocol to identify explicitly the toolkit applet for which the message is sent;
- unformatted (e.g. a toolkit applet specific protocol) then the (U)SAT Framework will pass this data to all registered toolkit applets.

When the Short Message is received as Concatenated Short Messages as defined in TS 23.040[10], it is the responsibility of the (U)SAT Framework to link single Short Messages together to re - assemble the original message before any further processing. The original Short Message shall be placed in one SMS TPDU TLV (with TP-UDL field coded on one octet) included in the *USATEnvelopeHandler*. The concatenation control headers used to re-assemble the short messages in the correct order shall not be present in the SMS TPDU. 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).

The minimum requirement for the (U)SAT Framework is to process a concatenated short message with the following properties:

- the Information Element Identifier is equal to the 8-bit reference number.
- it contains uncompressed 8 bit data or uncompressed UCS2 data.

EVENT_FORMATTED_SMS_PP_ENV

Upon reception of a TS 31.115[9] formatted Short Message Point to Point (Single or Concatenated) via an ENVELOPE, the (U)SAT Framework shall:

- verify the security of the Short Message as per TS 31.115[9];
- trigger the toolkit applet registered with the corresponding TAR;
- take the optional Application Data posted by the triggered toolkit applet if present;

- secure and send the response packet using SMS-DELIVER-REPORT or SMS-SUBMIT.

When the toolkit applet is triggered, data shall be provided deciphered.

EVENT_UNFORMATTED_SMS_PP_ENV

Upon reception of an unformatted Short Message Point to Point (Single or Concatenated) via an ENVELOPE, the (U)SAT Framework shall trigger all the Toolkit Applets registered to this event.

Note: As a consequence of the *EnvelopeResponseHandler* availability rules specified in clause 6.6, only the first triggered toolkit applet is guaranteed to be able to send back a response.

EVENT_FORMATTED_SMS_PP_UPD

Upon reception of a TS 31.115[9] formatted Short Message Point to Point (Single or Concatenated) via an UPDATE RECORD EF_{SMS}, the (U)SAT Framework shall:

- update the EF_{SMS} file with the data received, it is then up to the receiving toolkit applet to change the SMS stored in the file (i.e. the toolkit applet need to have access to the EF_{SMS} file)
- verify the security of the Short Message as per TS 31.115[9];
- convert the UPDATE RECORD EF_{SMS} APDU into a COMPREHENSION TLV List;
- trigger the toolkit applet registered with the corresponding TAR;

When the toolkit applet is triggered, data shall be provided deciphered.

The *USATEnvelopeHandler* provided to the applet shall:

- return *BTAG_SMS_PP_DOWNLOAD* to the *getTag()* method call;
- return the Comprehension TLV list length to the *getLength()* method call;

The *USATEnvelopeHandler* provided to the applet shall contain the following COMPREHENSION TLVs :

- Device Identities TLV

The Device Identities Comprehension TLV is used to store the information about the absolute record number in the EF_{SMS} file and the value of the EF_{SMS} record status byte, and is formatted as defined below:

Device identities Comprehension TLV
Device Identities tag
length = 02
Absolute Record Number
Record Status

With the absolute record number the toolkit applet can update EF_{SMS} in absolute mode to change the received SMS (e.g. in a readable text).

For Concatenated Short Message the Absolute Record Number and the Record Status will correspond to the last UPDATE RECORD EF_{SMS} APDU received.

- Address TLV

The value is the TS-Service-Centre-Address (RP-OA) of the last UPDATE RECORD EF_{SMS} APDU.

- SMS TPDU TLV

The value is the SMS TPDU provided deciphered and reassembled, if needed

- AID TLV

The AID comprehension TLV is present only if the EF_{SMS} file updated is under an ADF. The value is the AID of the ADF as defined TS 102 223[6].

The order of the TLVs given in the *USATEnvelopeHandler* is not specified,

Note: To get each COMPREHENSION TLV, it is recommended that the applet uses the *ViewHandler.findTLV()* methods

EVENT_UNFORMATTED_SMS_PP_UPD

Upon reception of an unformatted Short Message Point to Point (Single or Concatenated) via UPDATE RECORD EF_{SMS} APDU, the (U)SAT Framework shall :

- update the EF_{SMS} file with the data received;
- convert the UPDATE RECORD EF_{SMS} APDU data into a COMPREHENSION TLV List (as described for *EVENT_FORMATTED_SMS_PP_UPD*);
- trigger all the Toolkit Applets registered to this event.

The content of EF_{SMS} may have been modified by a previously triggered Toolkit Applet..

EVENT_FORMATTED_SMS_CB, EVENT_UNFORMATTED_SMS_CB

The received Cell Broadcast Message, via an ENVELOPE (CELL BROADCAST DOWNLOAD) APDU as defined in TS 31.111[7] and TS 51.014[8] and, can be either:

- formatted according to TS 31.115 [9] or an other protocol to identify explicitly the toolkit applet for which the message is sent;
- unformatted (e.g. using a toolkit applet specific protocol), then the (U)SAT Framework will pass this data to all registered toolkit applets.

When the Cell Broadcast Message is received as multiple pages as defined in TS 23.041[5], it is the responsibility of the (U)SAT Framework to link single pages together to re-assemble the original message before any further processing. The original Cell Broadcast message shall be placed in one Cell Broadcast page TLV included in the *USATEnvelopeHandler*. The message parameters shall correspond to the ones in the last received Cell Broadcast page (independently of the Page Parameter).

EVENT_FORMATTED_SMS_CB

Upon reception of a TS 31.115[9] formatted Cell Broadcast message, the (U)SAT Framework shall:

- verify the security of the Cell Broadcast message as per TS 31.115[9];
- trigger the toolkit applet registered with the corresponding TAR;

When the toolkit applet is triggered, data shall be provided deciphered.

EVENT_UNFORMATTED_SMS_CB

Upon reception of an unformatted Cell Broadcast message, the (U)SAT Framework shall trigger all the Toolkit Applets registered to this event.

EVENT_MO_SHORT_MESSAGE_CONTROL_BY_NAA

Upon reception of an ENVELOPE (MO SHORT MESSAGE CONTROL defined in TS 51.014[8] and TS 31.111[7]) APDU as defined in TS 102 221[6] and TS 51.011[4] the (U)SAT Framework shall trigger the Toolkit Applet registered to this event. The (U)SAT Framework shall not allow more than one Toolkit Applet to be registered to this event at a time(e.g. if a Toolkit Applet is registered to this event but not in selectable state the (U)SAT Framework shall not allow another Toolkit Applet to register to this event).

EVENT_FORMATTED_USSD, EVENT_UNFORMATTED_USSD

The received USSD String, via an ENVELOPE (USSD Data Download) APDU as defined in TS 31.111 [7], may be:

- formatted according to TS 31.115 [9] or an other protocol to identify explicitly the toolkit applet for which the message is sent;
- unformatted (e.g. a toolkit applet specific protocol) then the (U)SAT Framework will pass this data to all registered toolkit applets.

When the USSD Message is received as concatenated as defined in TS 31.115 [9], it is the responsibility of the (U)SAT Framework to link single USSD Messages together to re-assemble the original message before any further processing. The original USSD message shall be placed in one USSD String TLV included in the *USATEnvelopeHandler*. The USSD String parameters (DCS, PFI, CCF) shall correspond to the ones in the last received USSD String (independently of the CCF Sequence number).

EVENT_FORMATTED_USSD

Upon reception of a TS 31.115 [9] formatted USSD Message via an ENVELOPE, the (U)SAT Framework shall:

- verify the security of the USSD Message as per TS 31.115 [9];
- trigger the toolkit applet registered with the corresponding TAR;
- take the optional Application Data posted by the triggered toolkit applet if present;
- secure and send the response packet.

When the toolkit applet is triggered, data shall be provided deciphered.

EVENT_UNFORMATTED_USSD

Upon reception of an unformatted USSD String via an ENVELOPE, the (U)SAT Framework shall trigger all the Toolkit Applets registered to this event.

Note: As a consequence of the *EnvelopeResponseHandler* availability rules specified in clause 6.6, only the first triggered toolkit applet is guaranteed to be able to send back a response.

The following events defined in TS 102 221[6] shall be raised upon reception of the corresponding APDU defined in either TS 51.011[4] or TS 102 221[6].

EVENT_PROFILE_DOWNLOAD

EVENT_MENU_SELECTION, EVENT_MENU_SELECTION_HELP_REQUEST

EVENT_CALL_CONTROL_BY_NAA

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_STATUS_COMMAND

EVENT_EVENT_DOWNLOAD_LANGUAGE_SELECTION

EVENT_EVENT_DOWNLOAD_BROWSER_TERMINATION

EVENT_EVENT_DOWNLOAD_DATA_AVAILABLE

EVENT_EVENT_DOWNLOAD_CHANNEL_STATUS

EVENT_EVENT_DOWNLOAD_ACCESS_TECHNOLOGY_CHANGE

EVENT_EVENT_DOWNLOAD_DISPLAY_PARAMETER_CHANGED

EVENT_EVENT_DOWNLOAD_LOCAL_CONNECTION

EVENT_EVENT_DOWNLOAD_NETWORK_SEARCH_MODE_CHANGE

EVENT_EVENT_DOWNLOAD_BROWSING_STATUS

EVENT_PROACTIVE_HANDLER_AVAILABLE

EVENT_EXTERNAL_FILE_UPDATE

EVENT_FIRST_COMMAND_AFTER_ATR

EVENT_UNRECOGNIZED_ENVELOPE

6.3 Registration

A Toolkit Applet shall register to events described in 6.2 as defined in TS 102 241[2].

Constants for these events are available in *uicc.usim.toolkit.ToolkitConstants* interface in Annex A.

The *uicc.toolkit.ToolkitException* TAR_NOT_DEFINED shall be thrown if a Toolkit Applet has no TAR defined and registers to events: EVENT_FORMATTED_SMS_PP_ENV, EVENT_FORMATTED_SMS_PP_UPD, EVENT_FORMATTED_SMS_CB, EVENT_FORMATTED_USSD.

The *uicc.toolkit.ToolkitException*.EVENT_ALREADY_REGISTERED shall be thrown if there is another Toolkit Applet already registered to *EVENT_MO_SHORT_MESSAGE_CONTROL_BY_NAA.*

6.5 Envelope response handling

For the events defined in the present document, the following rules apply:

A Toolkit Applet can post a response by using the *post()* method or the *postAsBERTLV()* method defined in TS 102 241 [2]. The (U)SAT Framework shall return the Status Word as defined in TS 31.111 [7] and in TS 51.014 [8] depending on the current NAA.

Case of EVENT_MO_SHORT_MESSAGE_CONTROL_BY_NAA:

- The rules defined for *EVENT_CALL_CONTROL_BY_NAA* in TS 102 241 [2] apply.

Case of EVENT_UNFORMATTED_SMS_PP_ENV:

- See TS 102 241 [2].

Case of EVENT_FORMATTED_SMS_PP_ENV:

- When the *post()* or the *postAsBERTLV()* method is invoked, the (U)SAT Framework shall, according to bit 6 of the second octet of the SPI defined in TS 31.115 [9], build a SMS-DELIVER-REPORT or a SMS-SUBMIT.

In case of a SMS-DELIVER-REPORT, the (U)SAT Framework shall return the Status Word for RP-ACK or RP-ERROR as defined in TS 31.111 [7] and in TS 51.014 [8] depending on the current NAA.

In case of SMS-SUBMIT the boolean value method parameter shall be ignored by the (U)SAT Framework. If the SMS-SUBMIT is to be used, the (U)SAT Framework shall build and issue a Send Short Message proactive command as defined in TS 31.111 [7] and in TS 51.014 [8] depending on the current NAA .

Case of EVENT_FORMATTED_USSD:

- When the *post()* or the *postAsBERTLV()* method is invoked, the (U)SAT Framework shall build a USSD- String to be sent back in the Return Result Component contained in the subsequent Facility message. In that case the (U)SAT Framework shall return the Status Word as defined in TS 31.111 [7].

Case of EVENT_UNFORMATTED_USSD:

- See TS 102 241 [2].

6.6 System Handler management

For the handler management of the *ProactiveHandler*, the *ProactiveResponseHandler*, the *EnvelopeHandler* and the *EnvelopeResponseHandler*, the rules defined in TS 102 241[2] apply.

USATEnvelopeHandler:

The single system instance of the *USATEnvelopeHandler* and the single system instance of the *EnvelopeHandler* are two distinct objects instances.

- When available the *USATEnvelopeHandler* shall remain available and its content shall remain unchanged from the invocation to the termination of the *processToolkit()* method.
- The TLV List provided in the *USATEnvelopeHandler* are the same as in the *EnvelopeHandler*.
- The handler availability of the *USATEnvelopeHandler* is the same handler availability as the *EnvelopeHandler* including all the events defined in TS 102 241[2].

The following table describes the minimum availability of the handlers for all the events at the invocation of the *processToolkit()* method of the Toolkit Applet. The rules described in this table apply in addition to the rules described in "UICC API for Java Card™"

Table 2: Handler availability for each event

EVENT_	Reply busy allowed	EnvelopeHandler / USATEnvelopeHandler	EnvelopeResponse Handler	Nb of triggered / registrered Applet
_FORMATTED_SMS_PP_ENV	Y (see Note 1)	Y	Y	1 / n (per TAR)
_FORMATTED_SMS_PP_UPD	N	Y	N	1 / n (per TAR)
_UNFORMATTED_SMS_PP_ENV	Y	Y	Y	n / n
_UNFORMATTED_SMS_PP_UPD	N	Y	N	n / n
_FORMATTED_SMS_CB	Y	Y	N	1/n (per TAR)
_UNFORMATTED_SMS_CB	Y	Y	N	n / n
_MO_SHORT_MESSAGE_CONTROL_BY_NAA	N	Y	Y	1 / 1
_FORMATTED_USSD	Y	Y	Y	1 / n (per TAR)
_UNFORMATTED_USSD	Y	Y	Y	n / n

Note 1: The framework may reply busy and not trigger the toolkit applet if e.g. a PoR using SMS SUBMIT is required in the incoming message and a proactive session is ongoing.

Annex A

```
package uicc.usim.toolkit;


//import uicc.toolkit.ToolkitConstants;


/**
```

```
 *  <code>ToolkitConstants</code> encapsulates constants related to the USAT Toolkit applets.
 *
 * @version    2.0.0
 * @author     3GPP T3 API
 */
public interface ToolkitConstants extends uicc.toolkit.ToolkitConstants {
    /** Event : Envelope SMS-PP Data Download (31.115 formatted)  = 2       */
    public static final byte EVENT_FORMATTED_SMS_PP_ENV              = (short)2;
     /** Event : Update Record EF sms APDU     (31.115 formatted)  = 3       */
    public static final byte EVENT_FORMATTED_SMS_PP_UPD              = (short)3;
    /** Event : Envelope SMS-PP Data Download unformatted sms  = 4         */
    public static final byte EVENT_UNFORMATTED_SMS_PP_ENV            = (short)4;
    /** Event : Update Record EFsms APDU      unformatted sms   = 5        */
    public static final byte EVENT_UNFORMATTED_SMS_PP_UPD            = (short)5;
    /** Event : Cell Broadcast Data Download  = 6                        */
    public static final byte EVENT_UNFORMATTED_SMS_CB                = (short)6;
    /** Event : MO Short Message Control by SIM = 10                     */
    public static final byte EVENT_MO_SHORT_MESSAGE_CONTROL_BY_SIM   = (short)10;
    /** Event : Cell Broadcast Data Download Formatted = 24              */
    public static final byte EVENT_FORMATTED_SMS_CB                  = (short)24;
    /** Event : USSD Data Download Formatted = 121                */
    public static final byte EVENT_FORMATTED_USSD                = (short)121;
    /** Event : USSD Data Download Unformatted = 122              */
    public static final byte EVENT_UNFORMATTED_USSD              = (short)122;
}


//-------------------------------------------------------------------------------
// PACKAGE DEFINITION
//-------------------------------------------------------------------------------
package uicc.usim.toolkit;

//-------------------------------------------------------------------------------
// IMPORTS
//-------------------------------------------------------------------------------
```

```
import uicc.toolkit.ToolkitException;
import uicc.toolkit.EnvelopeHandler;
/**
 * The USATEnvelopeHandler interface contains basic methods to handle the <b>SMS Envelope
 * </b>data field. This interface will be used by the Toolkit applet in order to
 * have access to the current SMS Envelope information. No constructor is available
 * for the Toolkit applet.
 *
 * @author     T3 SWG API
 * @see        uicc.toolkit.EnvelopeHandler
 */
public interface USATEnvelopeHandler extends uicc.toolkit.EnvelopeHandler {


   /**
    * Looks for the Secured Data from the Command Packet in the first SMS TPDU
    * or Cell Broadcast Page Simple TLV or USSD String TLV contained in the Envelope handler.<br>
    * <br>
    * This can be used on the events:<ul>
    * <li>EVENT_FORMATTED_SMS_PP_ENV, EVENT_FORMATTED_SMS_PP_UPD, if the SMS TP-UD is formatted
    * according to TS 31.115 Single or Concatenated Short Message.
    * <li>EVENT_FORMATTED_SMS_CB, if the Cell Broadcast Page is formatted according to TS 31.115.
    * <li>EVENT_FORMATTED_USSD, if the USSD String is formatted according to TS 31.115.
    * If the element is available it becomes the TLV selected.</ul>
    * <br>
    * @return the offset of the Secured Data first byte in the first SMS TPDU or Cell Broadcast Page or USSD String TLV element.
    *        If the Secured Data length is zero the value returned shall be the offset of the first byte following
    *        the TS 31.115 Command Packet structure.
    *
    * @exception ToolkitException with the following reason codes: <ul>
    *      <li><code>UNAVAILABLE_ELEMENT</code> in case of unavailable SMS TPDU or Cell Broadcast Page TLV element or wrong data format </ul>
    */
   public short getSecuredDataOffset() throws ToolkitException;
```

```
   /**
    * Looks for the length of the Secured Data from the Command Packet in the first SMS TPDU
    * or Cell Broadcast Page Simple or USSD String TLV contained in the Envelope handler. <br>
    * <br>
    * This can be used on the events:<ul>
    * <li>EVENT_FORMATTED_SMS_PP_ENV, EVENT_FORMATTED_SMS_PP_UPD, if the SMS TP-UD
    * is formatted according to TS 31.115 Single or Concatenated Short Message.
    * <li>EVENT_FORMATTED_SMS_CB, if the Cell Broadcast Page is formatted according to TS 31.115.
    * <li>EVENT_FORMATTED_USSD, if the USSD String is formatted according to TS 31.115.
    * If the element is available it becomes the TLV selected.</ul>
    *
    * @return the length of the Secured Data contained in the first SMS TPDU or Cell Broadcast Page or USSD String TLV element (without padding bytes).
    *         If the Secured Data length is zero, no exception shall be thrown.
    *
    * @exception ToolkitException with the following reason codes: <ul>
    *      <li><code>UNAVAILABLE_ELEMENT</code> in case of unavailable SMS TPDU or Cell Broadcast Page or USSD String TLV element wrong data format </ul>
    */
   public short getSecuredDataLength() throws ToolkitException;


[…]

}
```