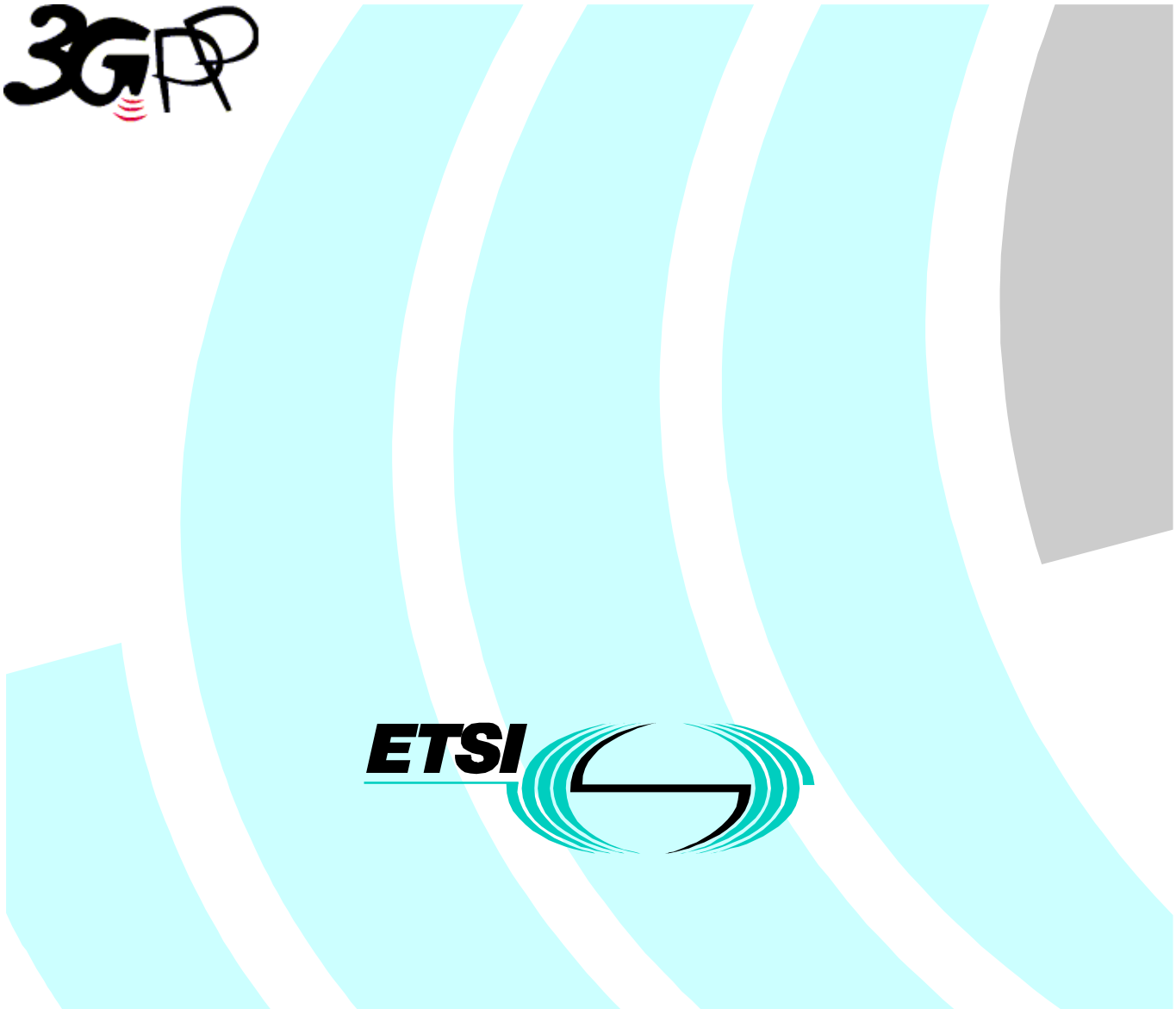
ETSI TR 129 998-5-1 V4.0.0 (2001-03)

Technical Report

Universal Mobile Telecommunications System (UMTS);
Open Service Access (OSA);
Application Programming Interface (API) Mapping for OSA;
Part 5: User Interaction Service Mapping;
Subpart 1: API to CAP Mapping
(3GPP TR 29.998-5-1 version 4.0.0 Release 4)



Reference
RTR/TSGN-0529998-5-1Uv4

Keywords
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://www.etsi.org/ipr).

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 the 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 www.etsi.org/key .

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

Structure of the OSA API Mapping (3GPP TR 29.998)

The Technical Report 3GPP TR 29.998 consists of a series of parts and subparts. An effort has been made to ensure that the part numbers used in the mapping TR correspond to the part numbers of the base OSA specification in 3GPP TS 29.198. For this reason, certain parts, for which no suitable mapping could be suggested, have not been delivered. At a later stage a mapping to a new protocol may become evident, in which case these missing parts will be developed.

The OSA documentation was defined jointly between 3GPP TSG CN WG5, ETSI SPAN 12 and the Parlay Consortium, in co-operation with the JAIN consortium. The 3GPP TR 29.998 is based on a mapping document with a wider scope, developed as part of this co-operation. Certain mappings defined in the course of this joint development are not applicable for 3GPP Release 4, which is why not all sub-parts have been delivered as part of 3GPP Release 4. However, it is expected that some will become applicable within the scope of 3GPP Release 5, which is why a common sub-part numbering is being retained, albeit with gaps for 3GPP Release 4.

If mapping for a certain Part is "Not Applicable" it can either indicate that a mapping does not exist (e.g. Part 2 Common Data), or the API is considered to be implemented directly on a physical entity, or via a proprietary mechanism.

The present document is part 5 subpart 1 of a multi-part TR covering the 3rd Generation Partnership Project: Technical Specification Group Core Network; Open Service Access (OSA); Application Programming Interface (API) Mapping for OSA, as identified below.

29.998-1: General Issues on API Mapping
29.998-2: Not Applicable
29.998-3: Not Applicable
29.998-4-1: Call Control Service Mapping; Subpart: API to CAP Mapping
29.998-4-2: Call Control Service Mapping; Subpart 2 generic call control INAP (not Rel-4)
29.998-4-3: Call Control Service Mapping; Subpart 3 multiparty call control INAP (not Rel-4)
29.998-4-4: Call Control Service Mapping; Subpart 4 multiparty call control SIP (not Rel-4)
29.998-4-5: Call Control Service Mapping; Subpart 5 multimedia call control extensions mapping to SIP (not Rel-4)
29.998-5-1: User Interaction Service Mapping; Subpart 1: API to CAP Mapping
29.998-5-2: User Interaction Service Mapping; Subpart 2 user interaction INAP (not Rel-4)
29.998-5-3: User Interaction Service Mapping; Subpart 3 user interaction Megacop (not Rel-4)
29.998-5-4: User Interaction Service Mapping; Subpart 4: API to SMS Mapping

29.998-6: User Location – User Status Service Mapping to MAP
29.998-7: Not Applicable
29.998-8: Data Session Control Service Mapping to CAP

OSA API specifications 29.198-family		OSA API Mapping - 29.998-family	
29.198-1	**Part 1: Overview**	**29.998-1**	**Part 1: Overview**
29.198-2	**Part 2: Common Data Definitions**	29.998-2	Not Applicable
29.198-3	**Part 3: Framework**	29.998-3	Not Applicable
29.198-4	**Part 4: Call Control SCF**	**29.998-4-1**	**Subpart 1: Generic Call Control – CAP mapping**
		29.998-4-2	
29.198-5	**Part 5: User Interaction SCF**	**29.998-5-1**	**Subpart 1: User Interaction – CAP mapping**
		29.998-5-2	
		29.998-5-3	
		29.998-5-4	**Subpart 4: User Interaction – SMS mapping**
29.198-6	**Part 6: Mobility SCF**	**29.998-6**	**User Status and User Location – MAP mapping**
29.198-7	**Part 7: Terminal Capabilities SCF**	29.998-7	Not Applicable
29.198-8	**Part 8: Data Session Control SCF**	**29.998-8**	**Data Session Control – CAP mapping**
29.198-9	Part 9: Generic Messaging SCF	29.998-9	Not Applicable
29.198-10	Part 10: Connectivity Manager SCF	29.998-10	Not Applicable
29.198-11	**Part 11: Account Management SCF**	29.998-11	Not Applicable
29.198-12	**Part 12: Charging SCF**	29.998-12	Not Applicable

1 Scope

The present document investigates how the OSA User Interaction Interface Class methods defined in 3GPP TS 29.198-5 [5] can be mapped onto CAMEL Application Part operations and Mobile Application Part operations. The mapping of the OSA API to the CAP and relevant MAP operations is considered informative, and not normative. An overview of the mapping TR is contained in the introduction of the present document as well as in 3GPP TR 29.998-1 [10].

The OSA specifications define an architecture that enables application developers to make use of network functionality through an open standardised interface, i.e. the OSA APIs. The API specification is contained in the 3GPP TS 29.198 series of specifications. An overview of these is available in the introduction of the present document as well as in 3GPP TS 29.198-1 [1]. The concepts and the functional architecture for the Open Service Access (OSA) are described by 3GPP TS 23.127 [3]. The requirements for OSA are defined in 3GPP TS 22.127 [2].

The present document has been defined jointly between 3GPP TSG CN WG5, ETSI SPAN 12 and the Parlay Consortium, in co-operation with the JAIN consortium.

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 29.198-1: "Open Service Access; Application Programming Interface; Part 1: Overview".

[2] 3GPP TS 22.127: "Stage 1 Service Requirement for the Open Service Access (OSA) (Release 4)".

[3] 3GPP TS 23.127: "Virtual Home Environment (Release 4)".

[4] 3GPP TR 22.905: "3GPP Vocabulary".

[5] 3GPP TS 29.198-5: "Open Service Access; Application Programming Interface - Part 5: Generic User Interaction".

[6] 3GPP TS 29.002: "Mobile Application Part (MAP) specification".

[7] 3GPP TS 29.078: "CAMEL Application Part (CAP) specification – Phase 3".

[8] 3GPP TS 22.101: "Universal Mobile Telecommunications System (UMTS): Service Aspects; Service Principles".

[9] ITU-T Q.850: "Usage of cause and location in the Digital Subscriber Signalling System No. 1 and the Signalling System No. 7 ISDN User Part".

[10] 3GPP TR 29.998-1: "API Mapping for Open Service Access; Part 1: General Issues on API Mapping".

3 Definitions and abbreviations

3.1 Definitions

For the purposes of the present document, the terms and definitions given in TS 29.198-1 [1] apply.

3.2 Abbreviations

For the purposes of the present document, the abbreviations given in TS 29.198-1 [1] apply.

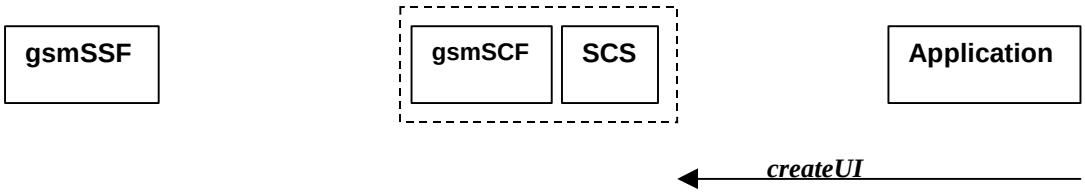
4 Generic Message Transfer Service CAMEL Call Flows

4.1 User Interaction

The User Interaction interface is used by applications to interact with end users. The API only supports Call User Interaction.

4.1.1 createUI

createUI is a method that is used to create a new (non call related) user interaction object.



Note: There are no associated CAP call flows

Figure 4-1: Call Flow for createUI

Table 4-1: Normal Operation

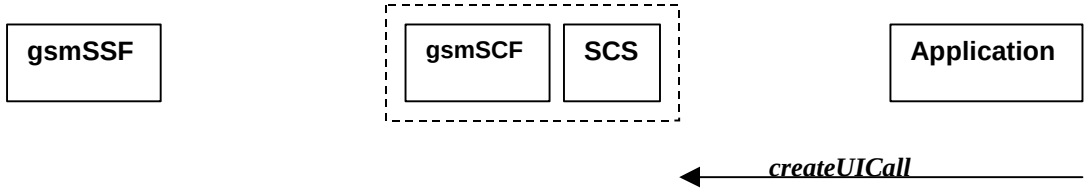
Pre-conditions	The application has been instructed to initiate a non call related User Interaction
1	The application invokes the ***createUI*** method
2	The SCS creates a new UI object

Parameter Mapping

None.

4.1.2 createUICall

createUICall is a method that is used to create a new call related user interaction object.



Note: There are no associated CAP call flows

Figure 4-2: Call Flow for createUICall

Table 4-2: Normal Operation

Pre-conditions	The application has been requested to initiate a call related User Interaction
1	The application invokes the ***createUICall*** method
2	The SCS creates a new ***UICall*** object

Parameter Mapping

None.

4.1.3 enableUINotification

enableUINotification is a method that enables the reception of a user initiated user interaction.

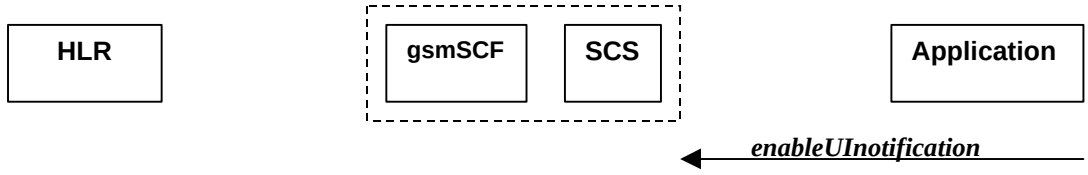


Figure 4-3: Call Flow for enableUINotification

Table 4-3: Normal Operation

Pre-conditions	An agreement is established between the network operator and the service provider for the event notification to be enabled
1	The application invokes the ***enableUINotification*** method
2	The SCS stores the request.

Parameter Mapping

None.

4.1.4 disableUINotification

disableUINotification is a method that allows the application to remove notification for UI related actions previously set.

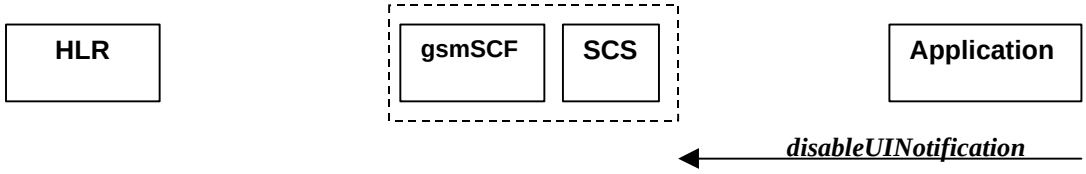


Figure 4-4: Call Flow for disableUINotification

Table 4-4: Normal Operation

Pre-conditions	An agreement is established between the network operator and the service provider for the event notification to be disabled
1	The application invokes the ***disableUINotification*** method
2	The request is disabled in the SCS.

Parameter Mapping

None.

4.1.5 userInteractionEventNotify

userInteractionEventNotify is a method that notifies the application of a user initiated request for user interaction.

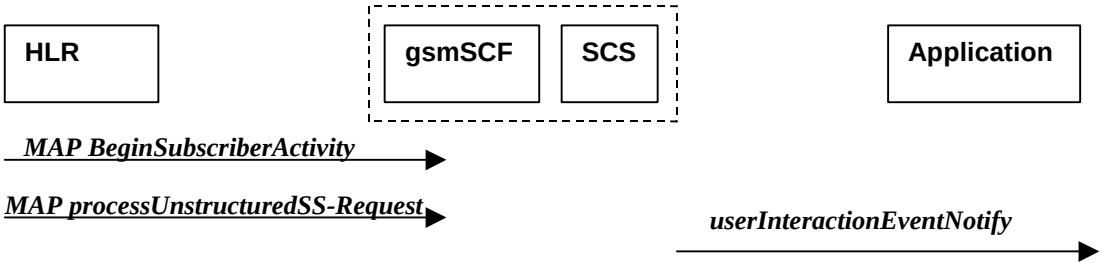


Figure 4-5: Call Flow for userInteractionEventNotify

Table 4-4: Normal Operation

Pre-conditions	
1	The gsmSCF receives a MAP ***processUnstructuredSS-Request*** message from the HLR. This operation may be preceded by MAP ***beginSubscriberActivity*** (see Note).
2	The gsmSCF sends an equivalent internal message to the SCS
3	The SCS identified the correct application that enable the notification request from the subscriber and invokes the ***userInteractionEventNotify*** method
NOTE: The MAP beginSubscriberActivity is sent in case of MAP version 1.	

Table 4-5: Parameter Mapping

From: processUnstructuredSS-Request	To: *userInteractionEventNotify*
	ui
	eventInfo (TpCallEventInfo) :
msisdn	OriginatingAddress
	DestinationAddress
	ServiceCode
	DataTypeIndication
ussd-DataCodingScheme ussd-String	DataString
	assignmentID
	appInterface (output)

4.1.6 userInteractionAborted

userInteractionAborted is a method that indicates to the application that the User Interaction service instance has terminated or closed abnormally. No further communication will be possible between the User Interaction service instance and the application.

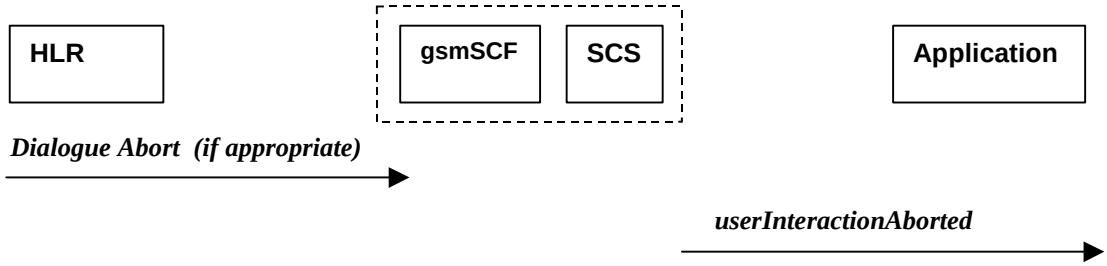


Figure 4-6: Call Flow for userInteractionAborted (scenario 1)

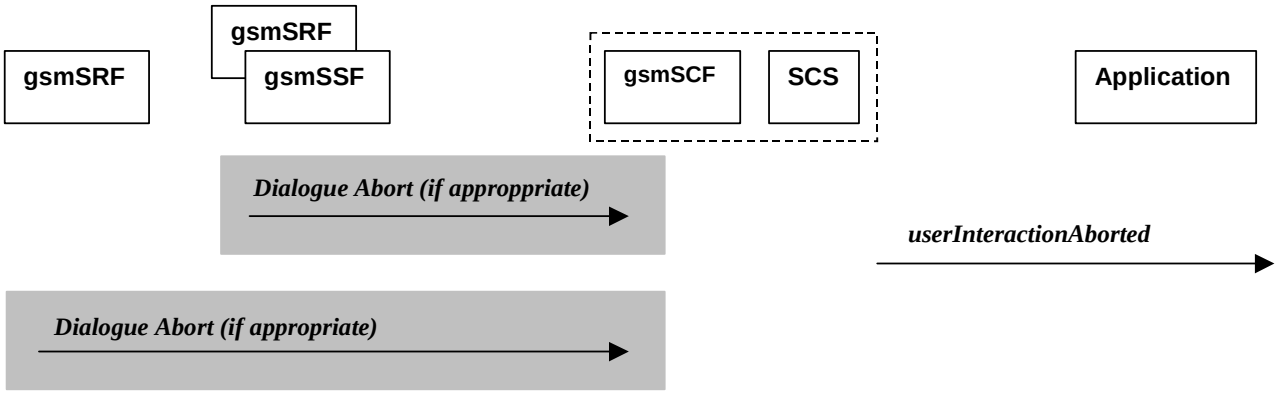


Figure 4-7: Call Flow for userInteractionAborted (scenario 2)

Parameter Mapping

None.

4.1.7 userInteractionNotificationInterrupted

userInteractionNotificationInterrupted is a method that indicates to the application that all user interaction event notifications have been temporarily interrupted.

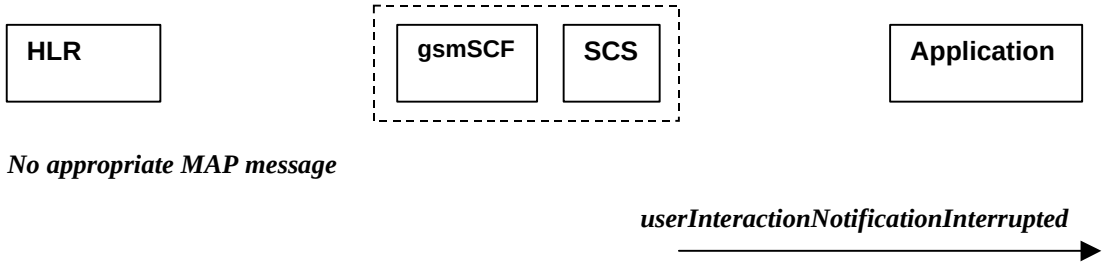


Figure 4-8: Call Flow for userInteractionNotificationInterrupted

Table 4-6: Normal Operation

Pre-conditions	User interaction event notifications have been enabled using the *enableUINotification* method on the UIManager interface
1	The SCS has detected, or has been informed of, a fault which prevents further user interaction events from being notified
2	The SCS invokes the ***userInteractionNotificationInterrupted*** method

Parameter Mapping

None.

4.1.8 userInteractionNotificationContinued

userInteractionNotificationContinued is a method that indicates to the application that user interaction event notifications will again be possible.

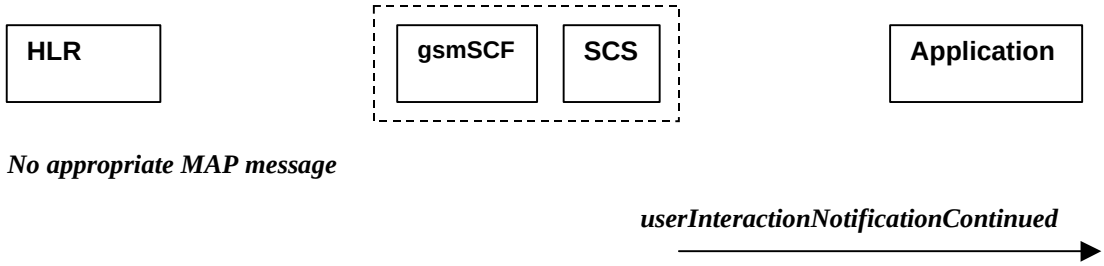


Figure 4-9: Call Flow for userInteractionNotificationContinued

Table 4-7: Normal Operation

Pre-conditions	User interaction event notifications have been interrupted and *userInteractionNotificationInterrupted* method has been invoked.
1	The SCS detects that user interaction event notifications are again possible.
2	The SCS invokes the ***userInteractionNotificationContinued*** method.

Parameter Mapping

None.

4.1.9 userInteractionFaultDetected

userInteractionFaultDetected is a method that indicates to the application that a fault has been detected in the user interaction. This method is invoked e.g. if the call has been deassigned.

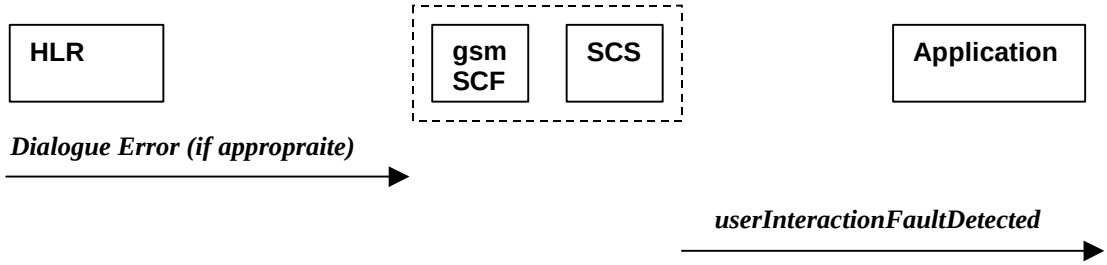


Figure 4-10: Call Flow for userInteractionFaultDetected (scenario 1)

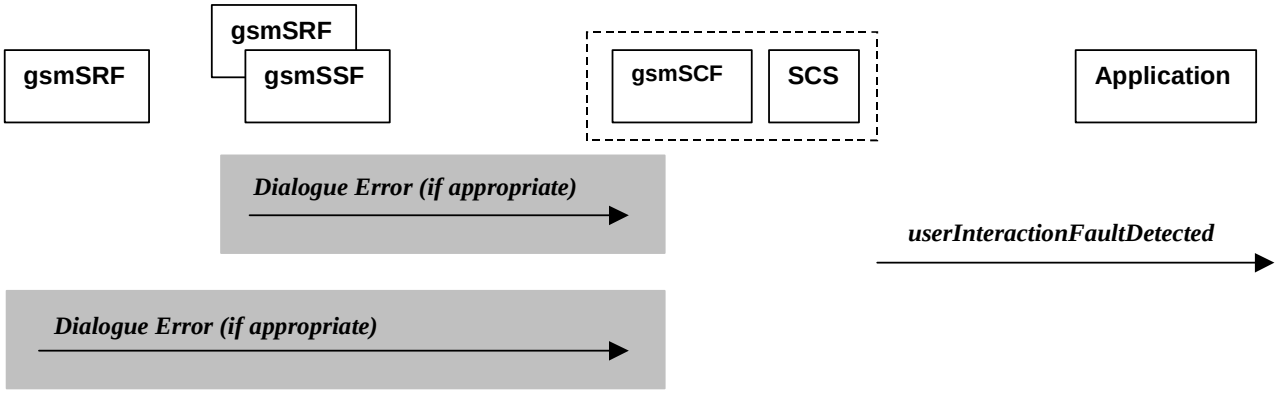


Figure 4-11: Call Flow for userInteractionFaultDetected (scenario 2)

Table 4-8: Normal Operation

Three Alternatives have been identified

1. USSD based interaction between the MS and the gsmSCF

Pre-conditions	USSD interaction is in progress and a dialogue is running between the HLR and gsmSCF
1	The gsmSCF detects or receives an indication that the there is an error in the user interaction
2	The gsmSCF sends an equivalent internal message to the SCS
3	The SCS invokes the ***userInteractionFaultDetected*** method to the appropriate application

Table 4-9:

2. Interaction between a gsmSRF internal to the gsmSSF and the gsmSCF

Pre-conditions	User interaction is in progress between the gsmSRF and the gsmSCF
1	The gsmSCF detects or receives an indication that there is an error in the user interaction
2	The gsmSCF sends an equivalent internal message to the SCS
3	The SCS invokes the ***userInteractionFaultDetected*** method to the appropriate application

Table 4-10:

3. Interaction between a gsmSRF and the gsmSCF

Pre-conditions	User interaction is in progress between the gsmSRF and the gsmSCF
1	The gsmSCF detects or receives an indication that the there is an error in the user interaction
2	The gsmSCF sends an equivalent internal message to the SCS
3	The SCS invokes the ***userInteractionFaultDetected*** method to the appropriate application

Table 4-11: Parameter Mapping

From: Dialogue Error	To: *userInteractionFaultDetected*
	userInteractionIdentifier
	fault
ReturnError	

4.1.10 sendInfoReq

sendInfoReq is an asynchronous method that sends information to the user.

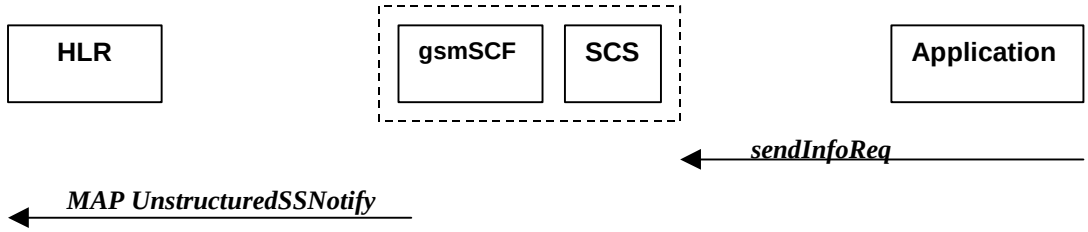


Figure 4-12: Call Flow for sendInfoReq (scenario 1)

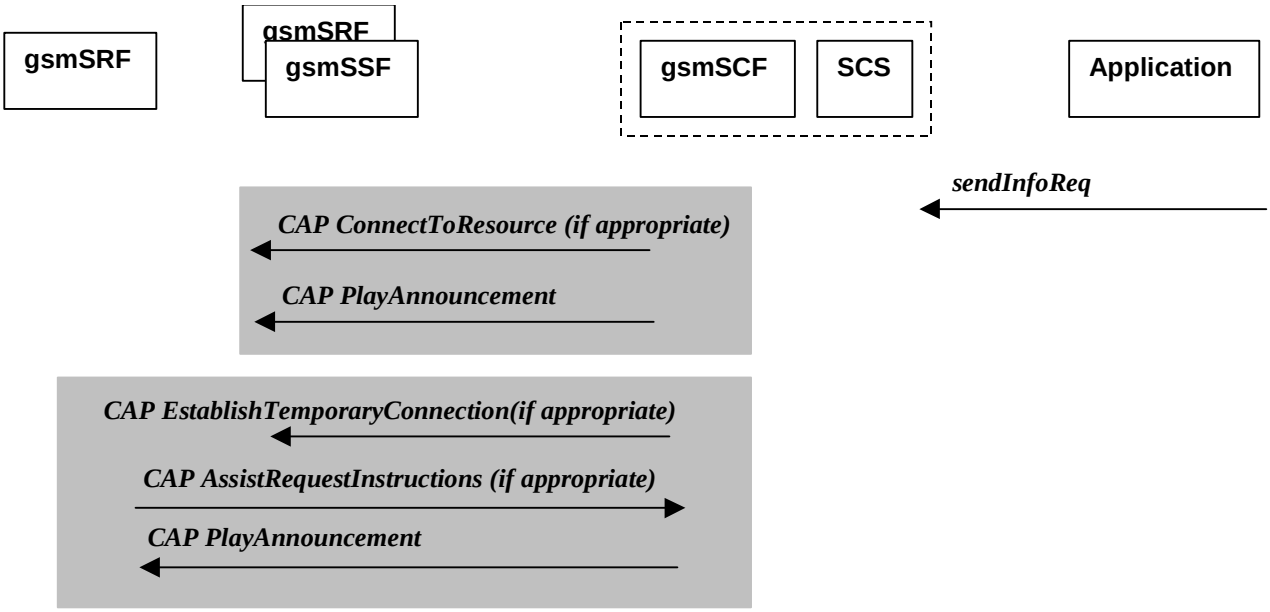


Figure 7-13: Call Flow for sendInfoReq (scenario 2)

Table 4-12: Normal Operation

Three Alternatives have been identified

1. USSD based interaction between the MS and the gsmSCF

Pre-conditions	USSD interaction
1	The application invokes the sendInfo method
2	The SCS sends an equivalent internal message to the gsmSCF
3	The gsmSCF sends a MAP ***UnstructuredSS-Notify*** message to the HLR. If processUnstructuredSS-Request was previously received its result component may be sent containing ussd-DataCodingScheme and ussd-String. Note : For call-related USSD cases, the USSD is sent to the calling party.

Table 4-13:

2. Interaction between a gsmSRF internal to the gsmSSF and the gsmSCF

Pre-conditions	
1	The application invokes the ***sendInfoReq*** method
2	The SCS sends an equivalent internal message to the gsmSCF
3	The gsmSCF is aware of a gsmSRF internal to the gsmSSF. The gsmSCF sends CAP ***ConnectToResource***, and CAP ***PlayAnnouncement*** messages the gsmSSF Note : The user interaction shall apply to all parties connected to the call segment for the user interactions initiated by the connectToResource and establishTemporaryConnection operations.

Table 4-14:

3. Interaction between a gsmSRF internal to the gsmSSF and the gsmSCF

Pre-conditions	
1	The application invokes the ***sendInfoReq*** method
2	The SCS sends an equivalent internal message to the gsmSCF
3	The gsmSCF is aware of an external gsmSRF. The gsmSCF sends CAP ***EstablishTemporaryConnection*** message the gsmSSF.
4	On receipt of the CAP ***AssistRequestInstructions*** message from the gsmSRF, the gsmSCF sends the CAP ***PlayAnnouncement*** message to the gsmSRF. Note : The user interaction shall apply to all parties connected to the call segment for the user interactions initiated by the ***connectToResource*** and ***establishTemporaryConnection*** operations.

Table 4-15: Parameter Mapping

From: sendInfoReq	To: MAP unstructuredSS-Notify
userInteractionSessionID	
info (choice)	
infoID	
infoData	ussd-DataCodingScheme ussd-String
infoAddress	
variableInfoSet	
repeatIndicator	
responseRequested	
assignmentID	
	alertingPattern
	msisdn

Table 4-16:

From: sendInfoReq	To: CAP PlayAnnouncement
userInteractionSessionID	
info (choice) infoID	InformationToSend (choice) inbandInfo messageID (choice) elementaryMessageID text messageContent attributes elementaryMessageIDs variableMessage elementaryMessageID variableParts (sequence of the following choices) integer number time date price numberOfRepetitions duration interval tone toneID duration
infoData	
infoAddress	
variableInfoSet	The contents are directly mapped to variableParts above
variablePartInteger	
variablePartAddress	
variablePartTime	
variablePartDate	
variablePartPrice	
repeatIndicator	This integer value is directly mapped to numberOfRepetitions above
	disconnectFromIPForbidden (according to responseRequested)
responseRequested	requestAnnouncementComplete
assignmentID	

4.1.11 sendInfoRes

sendInfoRes is an asynchronous method that informs the application about the start or the completion of a ***sendInfoReq()***. This response is called only if the application has requested a response.

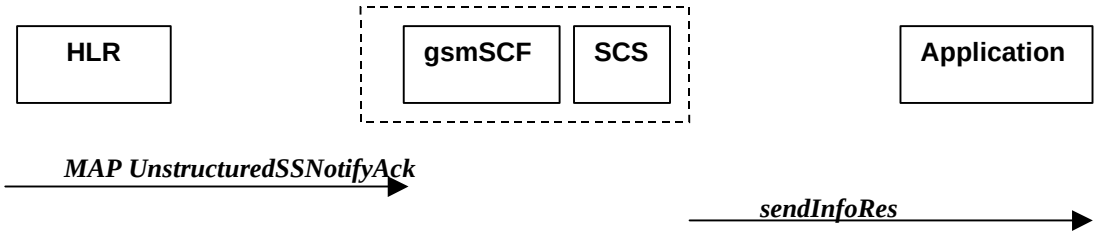


Figure 4-14: Call Flow for sendInfoRes (scenario 1)

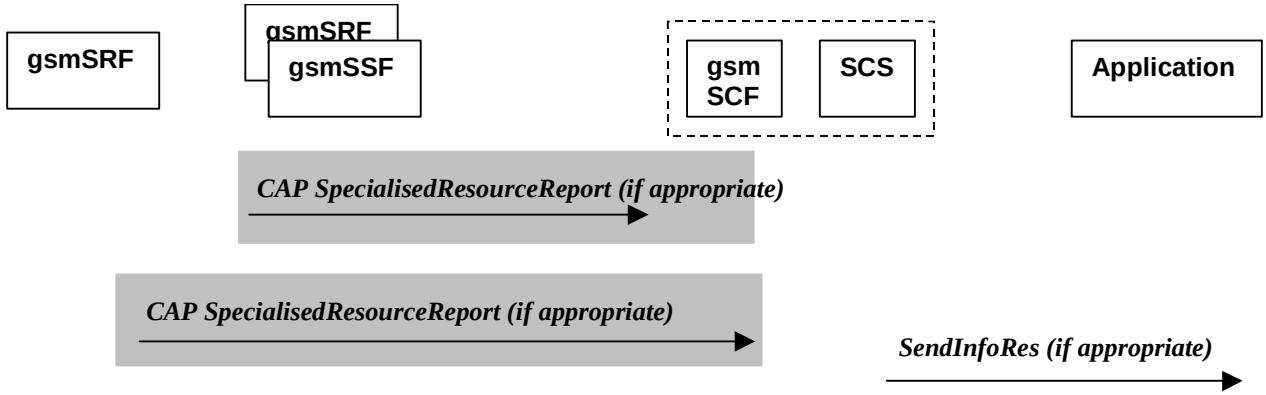


Figure 4-15: Call Flow for sendInfoRes (scenario 2)

Table 4-17: Normal Operation

Three Alternatives have been identified

1. USSD based interaction between the MS and the gsmSCF

Pre-conditions	The application has previously invoked the *sendInfoReq* method and has requested a notification
1	The gsmSCF receives an MAP unstructured***SS-Notify acknowledgement*** message from the HLR
2	The gsmSCF sends an equivalent internal message to the SCS
3	The SCS identifies the correct application and invokes the ***sendInfoRes*** method

Table 4-18:

2. Interaction between a gsmSRF internal to the gsmSSF and the gsmSCF

Pre-conditions	The application has previously invoked the *sendInfoReq* method and has requested a notification
1	The gsmSCF receives a CAP ***SpecialisedResourceReport*** message from the gsmSSF indicating that the announcement has been played to the subscriber
2	The gsmSCF sends an equivalent internal message to the SCS
3	The SCS identifies the correct application and invokes the ***sendInfoRes*** method

Table 4-19:

3. Interaction between a gsmSRF internal to the gsmSSF and the gsmSCF

Pre-conditions	The application has previously invoked the *sendInfoReq* method and has requested a notification
1	The gsmSCF receives a CAP ***SpecialisedResourceReport*** message from the gsmSRF indicating that the announcement has been played to the subscriber
2	The gsmSCF sends an equivalent internal message to the SCS
3	The SCS identifies the correct application and invokes the ***sendInfoRes*** method

Table 4-20: Parameter Mapping

From: CAP SpecialisedResourceReport	To: sendInfoRes
	userInteractionSessionID
	assignmentID
	response

4.1.12 sendInfoErr

sendInfoErr is an asynchronous method that indicates that the request to send information was unsuccessful.

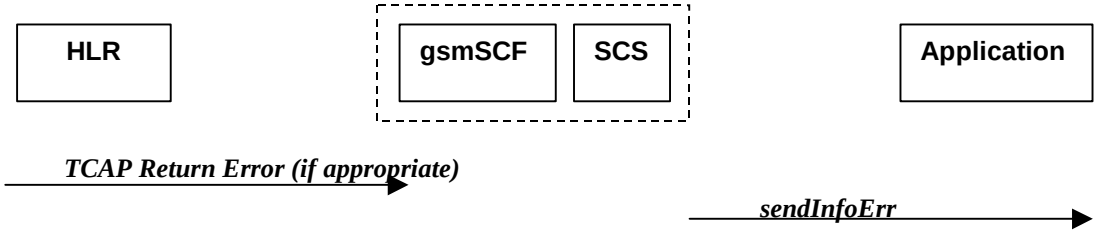


Figure 4-16: Call Flow for sendInfoErr (scenario 1)

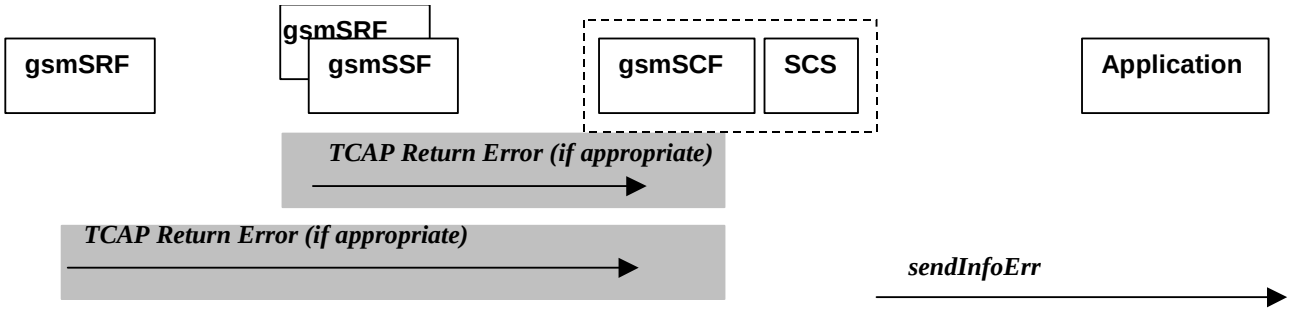


Figure 4-17: Call Flow for sendInfoErr (scenario 2)

Table 4-21: Normal Operation

For:

1. USSD based interaction between the MS and the CSE

2. Interaction between a gsmSRF internal to the gsmSSF and the CSE

3. Interaction between a gsmSRF internal to the gsmSSF and the CSE

Pre-conditions	The application has previously invoked the *sendInfoReq* method
1	The gsmSCF receives a message from the either the HLR, the gsmSSF or the gsmSRF indicating an error in the previous ***sendInfoReq*** method. Alternatively the gsmSCF may internal detect that the application has incorrectly sent the information
2	The gsmSCF sends an equivalent internal message to the SCS
3	The SCS identifies the correct application and invokes the ***sendInfoErr*** method

Table 4-22: Parameter Mapping

From: TCAP Return Error	To: sendInfoErr
	userInteractionSessionID
InvokeID	assignmentID
Error	error

4.1.13 sendInfoAndCollectReq

sendInfoAndCollectReq is an asynchronous method that plays an announcement or sends other information to the user and collects some information from the user. The announcement usually prompts for a number of characters (for example, these are digits or text strings such as "YES" if the user's terminal device is a phone).

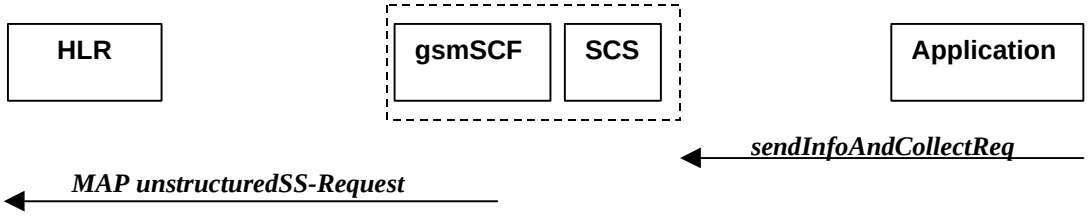


Figure 4-18: Call Flow for sendInfoAndCollectReq (scenario 1)

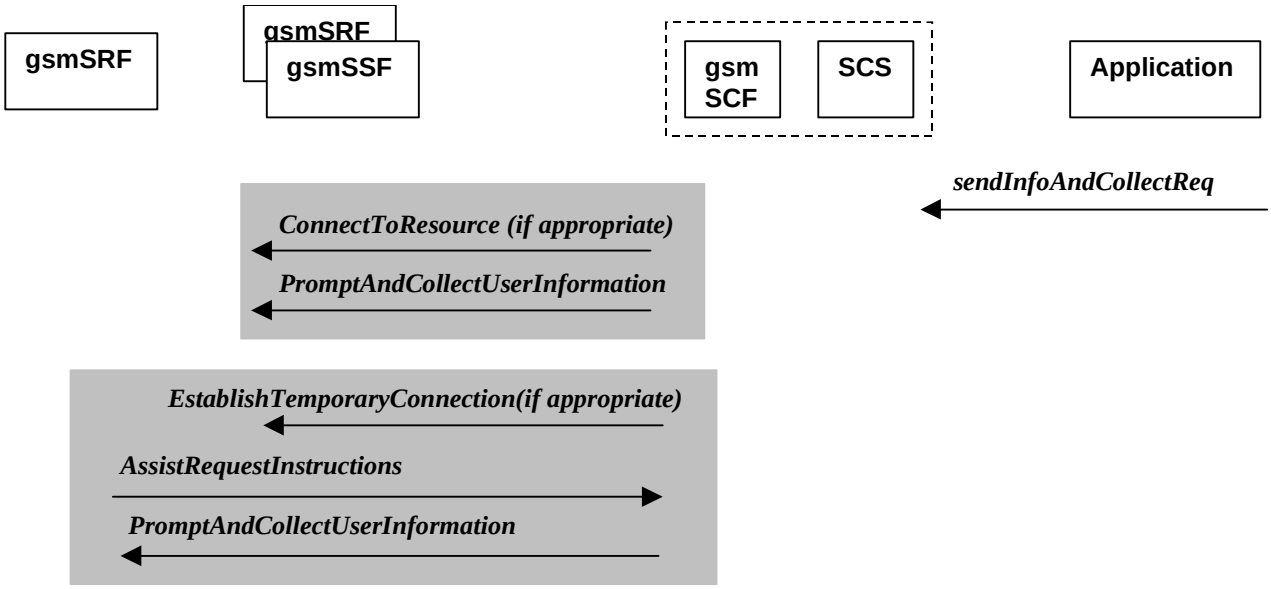


Figure 4-19: Call Flow for sendInfoAndCollectReq (scenario 2)

Table 4-23: Normal Operation

Three Alternatives have been identified

1. USSD based interaction between the MS and the gsmSCF

Pre-conditions	USSD interaction
1	The application invokes the ***sendInfoAndCollectReq*** method
2	The SCS sends an equivalent internal message to the gsmSCF
3	The gsmSCF sends a MAP ***unstructuredSS-Request message.***

Table 4-24:

2. Interaction between a gsmSRF internal to the gsmSSF and the gsmSCF

Pre-conditions	
1	The application invokes the ***sendInfoAndCollectReq*** method
2	The SCS sends an equivalent internal message to the gsmSCF
3	The gsmSCF is aware of a gsmSRF internal to the gsmSSF. The gsmSCF sends CAP ***ConnectToResource*** and ***PromptAndCollectUserInformation*** messages the gsmSSF

Table 4-25:

3. Interaction between a gsmSRF internal to the gsmSSF and the gsmSCF

Pre-conditions	
1	The application invokes the ***sendInfoAndCollectReq*** method
2	The SCS sends an equivalent internal message to the gsmSCF
3	The gsmSCF is aware of an external gsmSRF. The gsmSCF sends CAP ***EstablishTemporaryConnection***, message the gsmSSF.
4	On receipt of the CAP ***AssistRequestInstructions*** message from the gsmSRF, the gsmSCF sends the CAP ***PromptAndCollectUserInformation*** message to the gsmSRF

Table 4-26: Parameter Mapping

From: sendInfoAndCollectReq	To: MAP unstructuredSS-Request
userInteractionSessionID	
info (choice)	
infoID	
infoData	ussd-DataCodingScheme ussd-String
infoAddress	
variableInfo	
criteria	
responseRequested	
	alertingPattern
	msisdn
assignmentID	

Table 4-27:

From: sendInfoAndCollectReq	To: CAP PromptAndCollectUserInformation
userInteractionSessionID	
	disconnectFromIPForbidden (always true)
info (choice)	
infoID	InformationToSend (choice) inbandInfo messageID (choice) elementaryMessageID text messageContent attributes elementaryMessageIDs variableMessage elementaryMessageID variableParts (sequence of the following choices) integer number time date price numberOfRepetitions duration interval tone toneID duration
infoData	
infoAddress	
variableInfo	The contents are directly mapped to variableParts above
variablePartInteger	
variablePartAddress	
variablePartTime	
variablePartDate	
variablePartPrice	
criteria	collectedInfo
	collectedDigits
minLength	minimumNbOfDigits
maxLength	maximumNbOfDigits
endSequence	endOfReplyDigit
	cancelDigit
	startDigit
startTimeout	firstDigitTimeOut
interCharTimeout	interDigitTimeOut
	errorTreatment
	interruptableAnnInd
	voiceInformation
	voiceBack
responseRequested	
assignmentID	

4.1.14 sendInfoAndCollectRes

sendInfoAndCollectRes is an asynchronous method that returns the information collected to the application.

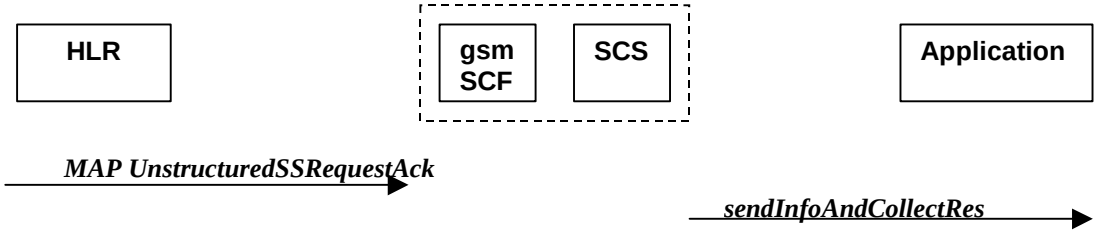


Figure 4-20: Call Flow for sendInfoAndCollectRes (scenario 1)

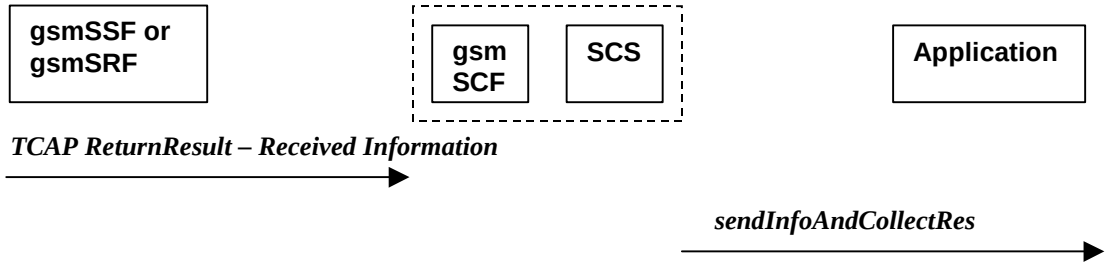


Figure 4-21: Call Flow for sendInfoAndCollectRes (scenario 2)

Table 4-28: Normal Operation

Two Alternatives have been identified

1. USSD based interaction between the MS and the gsmSCF

Pre-conditions	The application has invoked a *sendInfoAndCollectReq*()
1	The gsmSCF receives a MAP ***UnstructuredSS-Request acknowledgement*** message from the HLR
2	The gsmSCF sends an equivalent internal message to the SCS
3	The SCS invokes the ***sendInfoAndCollectRes*** method to the correct applications

Table 4-29:

2. Interaction with an gsmSRF internal to gsmSSF or external

Pre-conditions	The application has invoked a *sendInfoAndCollectReq*()
1	The gsmSCF receives a TCAP **ReturnResult** from the gsmSSF or the gsmSRF depending on whether a direct or indirect gsmSRF is used containing the Received Information.
2	The gsmSCF sends an equivalent internal operation to the SCS
3	The SCS identifies the correct application instance and invokes the ***sendinfoAndCollectRes*** method

Table 4-30: Parameter Mapping

From: MAP unstructuredSS-Request acknowledgement	To: sendInfoAndCollectRes
	userInteractionSessionID
	assignmentID
	response
ussd-DataCodingScheme ussd-String	info

Table 4-31:

From: TCAP Return Result (Received Information)	To: sendInfoAndCollectRes
	userInteractionSessionID
	assignmentID
	response
DigitsResponse	info (only the digits are mapped)

4.1.15 sendInfoAndCollectErr

sendInfoAndCollectErr is an asynchronous method that indicates that the request to send information and collect a response was unsuccessful.

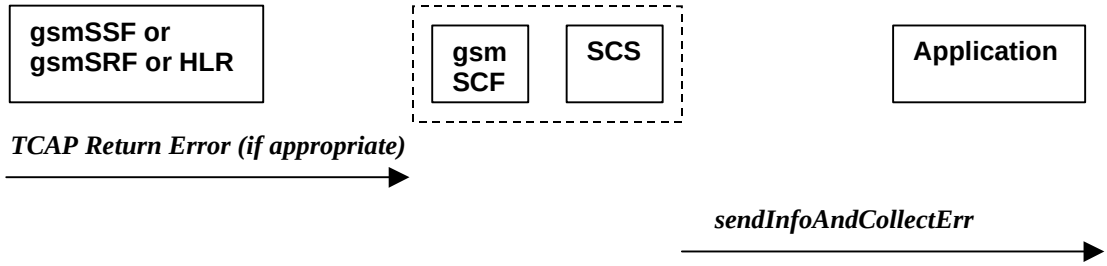


Figure 4-22: Call Flow for sendInfoAndCollectErr

Table 4-32:Normal Operation

Two Alternatives have been identified

1. USSD based interaction between the MS and the gsmSCF

Pre-conditions	The application has invoked a *sendInfoAndCollectReq*()
1	The gsmSCF detects an error in the ***sendInfoAndCollectReq*** method or receives a message form the HLR indicating an error that there is an error in ***sendInfoAndCollectReq*** method
2	The gsmSCF sends an equivalent internal message to the SCS
3	The SCS invokes the ***sendInfoAndCollectErr*** method to the correct application

Table 4-33:

2. Interaction with an gsmSRF internal to gsmSSF or external gsmSRF

Pre-conditions	The application has invoked a *sendInfoAndCollectReq*()
1	The gsmSCF either detects and error or receives a TCAP ***Error*** from the gsmSSF or the gsmSRF depending on whether a direct or indirect gsmSRF is used
2	The gsmSCF sends an equivalent internal operation to the SCS
3	The SCS identifies the correct application instance and invokes the ***sendInfoAndCollectErr*** method

Table 4-34: Parameter Mapping

From: TCAP Return Error	To: sendInfoAndCollectErr
	userInteractionSessionID
	assignmentID
error	error

4.1.16 release

release is a method that requests that the relationship between the application and the user interaction object be released. It causes the release of the used user interaction resources and interrupts any ongoing user interaction.

Call Flow

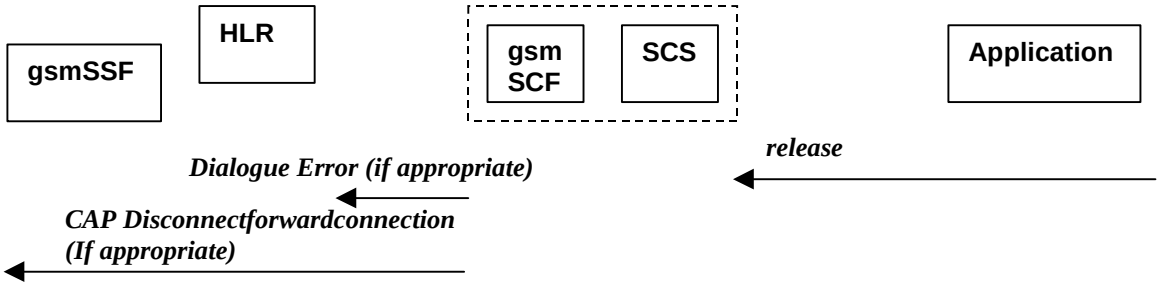


Figure 4-23: Call Flow for release

Table 4-35: Normal Operation

Two Alternatives have been identified

1. USSD based interaction

Pre-conditions	The gsmSCF has an open dialogue with the HLR
1	The application invokes a ***release***
2	The SCS sends an equivalent internal message to the gsmSCF
3	The gsmSCF sends a TCAP ***Abort*** message to the HLR if appropriate.

Table 4-36:

2. Interaction with a gsmSRF internal to gsmSSF or external gsmSRF

Pre-conditions	The application has previously invoked the *sendInfoAndCollectErr*. The gsmSCF is waiting for a response form the user
1	The application invokes a release
2	The SCS sends an equivalent internal message to the gsmSCF
3	The gsmSCF sends a CAP ***DisconnectForwardConnection*** to the gsmSSF

Table 4-37: Parameter Mapping

From: release	**To: Dialogue Error**
userInteractionSessionID	
	TC-U-ABORT TC-P-ABORT

Table 4-38:

From: release	**To: CAP** DisconnectForwardConnection
userInteractionSessionID	

4.1.17 abortActionReq

abortActionReq is an asynchronous method that aborts a user interaction operation, e.g. a ***sendInfoReq***, from the specified call. The call remains otherwise unaffected. The user interaction call service interrupts the current action on the specified call.

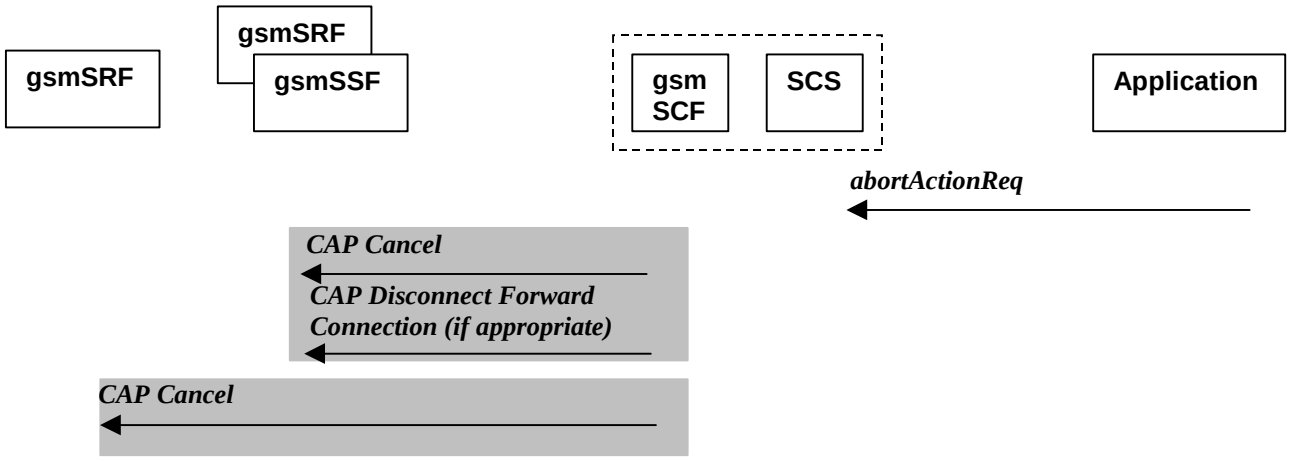


Figure 4-24: Call Flow for abortActionReq

Table 4-39: Normal Operation

Pre-conditions	The application has previously invoked e.g. the *sendInfoAndCollectReq*. The gsmSCF is waiting for a response form the user
1	The application invokes a ***abortActionReq***
2	The SCS sends an equivalent internal message to the gsmSCF
3	The gsmSCF sends a CAP ***Cancel*** message to the gsmSSF or the gsmSRF as appropriate and may send a CAP ***DisconnectForwardConnection*** to the gsmSSF if appropriate

Table 4-40: Parameter Mapping

From: abortActionReq	To: CAP Cancel
userInteractionSessionID	
assignmentID	InvokeID
	allRequests

4.1.18 abortActionRes

abortActionRes is an asynchronous method that confirms that the request to abort a user interaction operation on a call was successful.

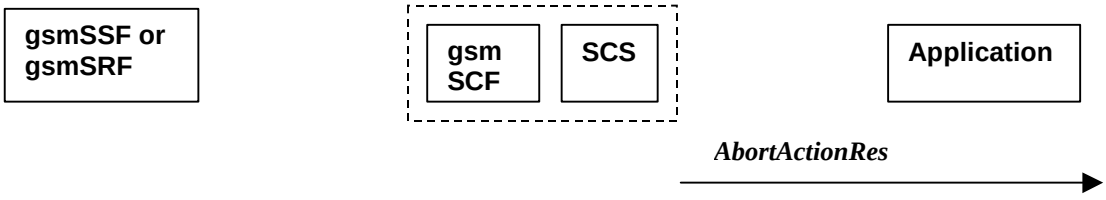


Figure 4-25: Call Flow for abortActionRes

There is no equivalent CAP/MAP mapping message

Table 4-41: Normal Operation

Pre-conditions	**The application has previously invoked the *abortActionReq*. The gsmSCF has sent the necessary instruction to the gsmSSF or the gsmSRF and is running a timer awaiting for any possible error return message. This timer expires and no errors are returned**
2	The gsmSCF determines that the CAP ***Cancel*** operation was successful. The gsmSCF sends an equivalent internal message to the SCS
3	The SCS invokes the ***abortActionRes*** method to the appropriate application.

4.1.19 abortActionErr

abortActionErr is an asynchronous method that indicates that the request to abort a user interaction on a call resulted in an error.

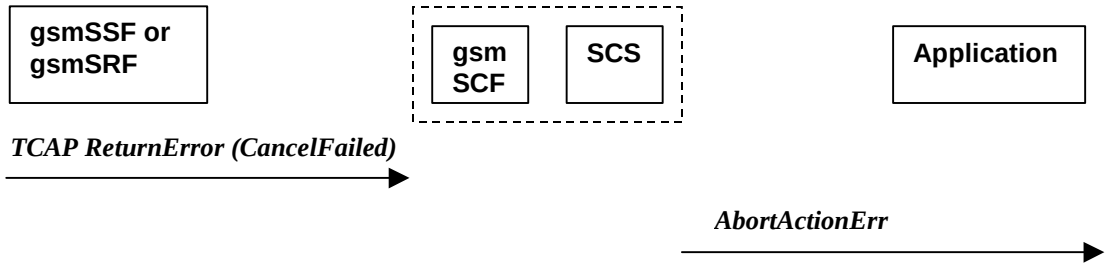


Figure 4-26: Call Flow for abortActionErr

Table 4-42: Parameter Mapping

From: TCAP error primitive	**To:** *abortActionErr*
	userInteractionSessionID
	assignmentID
TC-U-ERROR	error

Annex A (informative): Change history

Change history							
Date	**TSG #**	**TSG Doc.**	**CR**	**Rev**	**Subject/Comment**	**Old**	**New**
16 Mar 2001	CN_11	NP-010131	011	-	CR 29.998: for moving TR 29.998 from R99 to Rel 4 (N5-010159)	3.2.0	4.0.0

History

Document history		
V4.0.0	March 2001	Publication