WAP™ over GSM USSD

WAP-204-WAPOverGSMUSSD
Version 23-May-2000

Wireless Application Protocol
WAP over GSM USSD Specification

Disclaimer:

This document is subject to change without notice.

Contents

1. Scope

This document describes the mapping of WAP onto GSM Unstructured Supplementary Service Data (USSD) phase 2. The document includes a brief description of GSM USSD. For a complete description refer to [GSM0290], [GSM0390] and [GSM0490].

The GSM USSD service lack some fundamental services needed in order to use it as a bearer of the WAP protocols, and other applications with similar communication need. For example, the GSM USSD service is half-duplex and does not carry a destination address (only the MSISDN of the mobile phone). The USSD Dialogue Control Protocol (UDCP) has been defined in order to overcome these limitations. UDCP is specified in this document. The UDCP protocol is not part of the GSM Specifications from ETSI.

2. Document Status

This document is available online in the following formats:

- PDF format at http://www.wapforum.org/.

2.1 Copyright Notice

2.2 Errata

Known problems associated with this document are published at http://www.wapforum.org/.

2.3 Comments

Comments regarding this document can be submitted to the WAP Forum in the manner published at http://www.wapforum.org/.

3. References

3.1 Normative references

[GSM0290]	ETSI European Digital Cellular Telecommunication Systems (phase 2) : Unstructured Supplementary Service Data(USSD) - stage 1 (GSM 02.90)
[GSM0338]	ETSI European Digital Cellular Telecommunication Systems (phase 2+) : Alphabets and language-specific information (GSM 03.38)
[GSM0340]	ETSI European Digital Cellular Telecommunication Systems (phase 2+) : Technical realization of the Short Message Service (SMS) Point-to-Point (P) (GSM 03.40 version 5.6.x)
[GSM0390]	ETSI European Digital Cellular Telecommunication Systems (phase 2) : Unstructured Supplementary Service Data(USSD) - stage 2 (GSM 03.90)
[GSM0490]	ETSI European Digital Cellular Telecommunication Systems (phase 2) : Unstructured Supplementary Service Data(USSD) - stage 3 (GSM 04.90)
[GSM0902]	ETSI European Digital Cellular Telecommunication Systems (phase 2) : Mobile Application Part (MAP) specification (GSM 09.02)
[ISO8509]	ISO TR 8509 Service conventions
[RFC791]	Postel, J., “Internet Protocol”, RFC 791, September 1981.
[RFC2119]	S. Bradner “Keywords for use in RFCs to Indicate Requirement Levels”, RFC2119, http://www.internic.net/rfc/rfc2119.txt
[RFC2373]	Hinden, R. and S.Deering, “IP Version 6 Addressing Architecture”, RFC2373, July 1998.
[WDP]	"Wireless Datagram Protocol Specification", WAP Forum, 05-Nov-1999 http://www.wapforum.org/
[WINA]	"WAP Interim Naming Authority Process Document" http://www.wapforum.org/

4. Definitions and Abbreviations

4.1 Definitions

For the purpose of this specification the following definitions apply.

External Node
A node external to the GSM network. For example, a WAP Gateway.

Message
A general name for any PDU type.

Protocol Control Information (PCI)
Information exchanged between two protocol entities to coordinate their joint operation.

Protocol Data Unit (PDU)
A unit of data consisting of protocol control information and possibly user data.

Service Data Unit (SDU)
An amount of data handed down to the lower layer and whose identity is preserved from one end of a connection to the other.

Service Primitive
An abstract, implementation independent interaction between a user and the provider.

USSD Node
A node in the GSM network able to receive and initiate USSD dialogues. The network-node may be connected to the MSC, VLR or HLR. See [GSM0290].

USSD String
A parameter of the USSD operation. See [GSM0902].

USSD Operation
Operation in the GSM network used to send and receive USSD strings. See [GSM0902].

4.2 Abbreviations

For the purposes of this specification the following abbreviations apply.

DCS	Data Coding Scheme
ETSI	European Telecommunication Standardization Institute
GSM	Global System for Mobile Communication
IE	Information Element
MSISDN	Mobile Subscriber ISDN (Telephone number or address of device)
MS	Mobile Station
MSB	Most significant bits
NEI	Network Element Identifier
PCI	Protocol Control Information
PLMN	Public Land Mobile Network
RTT	Round-Trip Time
SAR	Segmentation and Reassembly
SMS	Short Message Service
PDU	Protocol Data Unit
SAP	Service Access Point
SDU	Service Data Unit
TCAP	Transaction Capability Application Part
UDCP	USSD Dialogue Control Protocol
UDH	User-Data Header (see [GSM 03.40])
UDHL	User-Data Header Length (see [GSM 03.40])
UDL	User-Data Length (see [GSM 03.40])
USSD	Unstructured Supplementary Service Data
WAE	Wireless Application Environment

WAP	Wireless Application Protocol
WCMP	Wireless Control Message Protocol
WSP	Wireless Session Protocol
WTP	Wireless Transaction Protocol
WDP	Wireless Datagram Protocol

4.3 Requirements

This specification uses the following words for defining the significance of each particular requirement:

MUST

This word, or the terms "REQUIRED" or "SHALL", mean that the definition is an absolute requirement of the specification.

MUST NOT

This phrase, or the phrase "SHALL NOT", mean that the definition is an absolute prohibition of the specification.

SHOULD

This word, or the adjective "RECOMMENDED", mean that there may exist valid reasons in particular circumstances to ignore a particular item, but the full implications must be understood and carefully weighed before choosing a different course.

SHOULD NOT

This phrase, or the phrase "NOT RECOMMENDED" mean that there may exist valid reasons in particular circumstances when the particular behaviour is acceptable or even useful, but the full implications should be understood and the case carefully weighed before implementing any behaviour described with this label.

MAY

This word, or the adjective "OPTIONAL", mean that an item is truly optional. One vendor may choose to include the item because a particular marketplace requires it or because the vendor feels that it enhances the product while another vendor may omit the same item. An implementation which does not include a particular option MUST be prepared to interoperate with another implementation which does include the option, though perhaps with reduced functionality. In the same vein an implementation which does include a particular option MUST be prepared to interoperate with another implementation which does not include the option (except, of course, for the feature the option provides.)

5. General description of USSD

5.1 Introduction

The GSM standard includes a wide range of supplementary services (Call Barring, Call Forwarding, etc). The services may be managed by entering text strings. For example, entering the text string "**21*1234567# SEND" will divert all incoming calls to the number 1234567. However, most MS manufacturers provide more user friendly methods to do this.

Normally when standard GSM supplementary services are managed from the MS structured, functional signaling is available. For example, if the user activates the supplementary service Call Forwarding the MS recognizes this and invokes a standard signaling procedure towards the network. This is because the Call Forwarding supplementary service is part of the GSM standard.

The supplementary services were introduced in stages into the GSM standard. To support old mobiles and Operator Specific Services, OSS, the container mechanism Unstructured Supplementary Service Data, USSD, was introduced into the GSM standard. The USSD operation can be used towards the network when the MS does not recognize the text string entered by the user. Further standardisation allowed the network to send USSD operations toward the MS, as well as combining mobile and network initiated operations in order to exchange data in a dialogue manner. Since the content of the operations sent from the network ends up on the MS display, and operations sent from the MS can be routed to an operator provided application in the network, USSD can be used as a transparent pipe through the GSM network.

The two most important features of USSD are:

1. It can be used by operators to provide operator specific services using a similar transport mechanism as when standard GSM supplementary services are used.
2. It can be used as a transparent bearer through the GSM network.

5.2 The USSD Standard

5.2.1 USSD phase 1

Network initiated operations are not supported, only mobile initiated. This means that the MS can send a request to the network and receive a response. There is no dialogue mechanism.

5.2.2 USSD phase 2

This is the present status of the standard. A dialogue is established between the mobile and the network-node. Multiple subsequent USSD operations can be sent within the dialogue.

5.2.3 USSD phase 2+ Enhanced USSD

The following USSD enhancements are considered by ETSI:

- Extending the usage of the DCS to distinguish between MMI-mode (DCS= '0000 1111') and bearer mode (DCS='01xx xxxx').
- In bearer mode: the DCS may indicate message classes; Immediate display, ME-specific, SIM-specific and TE-specific.
- In bearer mode: replacing the alphanumerical Service Code with a binary coded Network Element Identifier to be used when routing USSD through the network.
- In bearer mode: adapting the User Data Header concept from SMS. See [GSM0340].

- Support for multiple dialogues.

5.3 USSD Characteristics and Parameters

5.3.1 General

The following subchapters describe USSD specific characteristics and parameters.
USSD phase 2 is used as the baseline in the descriptions.

5.3.2 The USSD Dialogue

There are two types of USSD dialogues: mobile- and network initiated.

5.3.2.1 Mobile Initiated Dialogue

The mobile initiated dialogue is shown in table 1. The MS initiates the dialogue by invoking the ProcessUSSDRequest operation. The network can respond by either invoking a USSDRequest operation or release the dialogue by returning the result to the received ProcessUSSDRequest operation. Both the MS and the network can at any time release the dialogue by sending a RELEASE COMPLETE Radio Layer 3 message (END in TCAP).

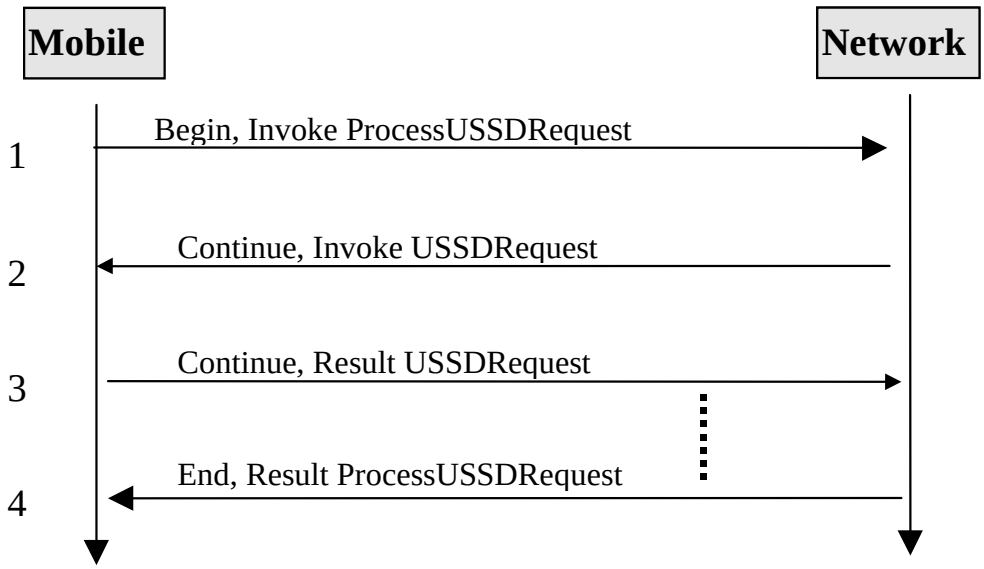


Figure1. Mobile initiated USSD dialogue.

5.3.2.2 Network Initiated Dialogue

The network initiated dialogue is shown in table 2. The network initiates the dialogue by invoking the USSDRequest operation. The MS responds by returning the result to the USSDRequest operation. Both the MS and the network can at any time release the dialogue by sending a RELEASE COMPLETE Radio Layer 3 message (END in TCAP).

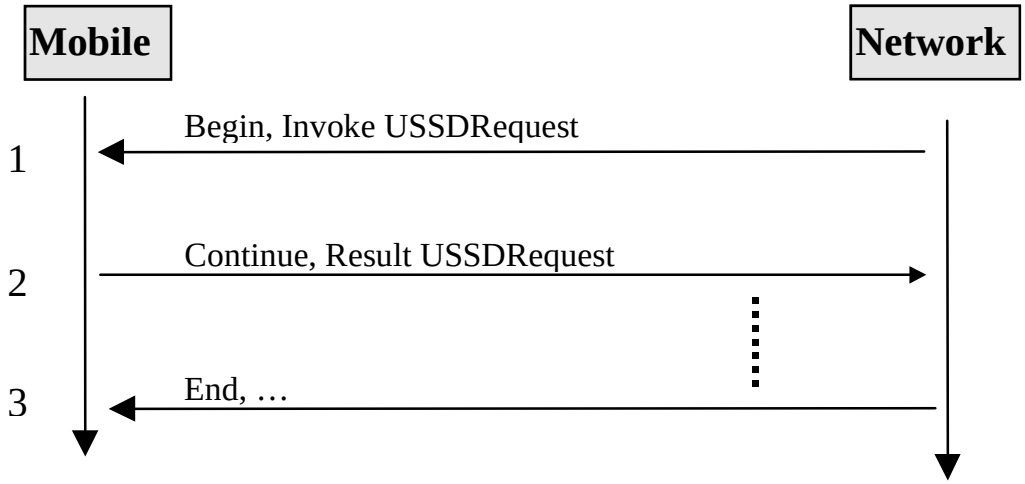


Figure 2. Network initiated USSD dialogue.

5.3.3 Data Coding Scheme (DCS)

A USSD operation has two parameters: the DCS and the USSD string. The DCS specifies the data coding scheme used in the USSD string. [GSM 0290] specifies the setting of the DCS for certain operations.

Table 1 Data Coding Scheme

Operation	**DCS setting as specified in [GSM0290] and coded according to [GSM0338]**
Mobile initiated operation	'Language unspecified' and 'SMS default alphabet'. DCS = 0000 1111
Mobile initiated operation response	Not specified.
Network initiated operation	Not specified.
Network initiated operation response	'Language unspecified' and 'SMS default alphabet'. DCS = 0000 1111

According to [GSM 0390] the network should reject the operation with an error if the DCS violates the standard setting.

5.3.4 Service Code (SC)

The service code is an in-heritage from the initially intended use of USSD as a mechanism to manage operator specific supplementary services (according to the MMI string format "*#SC*<data>#", in which SC is the service code).

The service code is part of the first USSD string sent from the MS and acts as a leader that guides the string to the end node (MSC, VLR or HLR depending on the value of the service code). Once a dialogue is established between the MS and the application a transparent pipe is opened through the network, thus the service code is not needed during the remaining part of the dialogue.

This procedure as well as the allocation of service codes is standardised. [GSM0290] identifies two types of service codes: VPLMN and HPLMN service codes; HPLMN service codes always routes the USSD string to the HLR, VPLMN service codes routes the string to the MSC/VLR.
To be able to use USSD from outside the GSM network, a USSD application located in the HLR has to relay the USSD string to an external node. The application has to provide the function:

F(service code) -> External Node Address

If the end application (the one that provides the service the user requested) is not located in the external node, the USSD string (extracted from the USSD operation and wrapped into another protocol) has to be passed on to the node that hosts the end application.

5.3.5 USSD Timers

To supervise USSD dialogues and to avoid hanging dialogues there are timers in the network. The timers are specified in [GSM 0902].

5.3.5.1 ProcessUSSDRequest Invoke Timer

The timer is started when the Invoke ProcessUSSDRequest is received by the network (the MS has initiated a dialogue). The timer runs until the Result ProcessUSSDRequest is sent to the MS (the dialogue is released).

This timer limits the total length of the dialogue. The value of the timer is between 1 and 10 minutes.

5.3.5.2 USSDRequest Invoke Timer

The timer is started when invoke USSDRequest is sent from the network and runs until the result USSDRequest is received.

This timer sets a restriction to the MS application processing time. For some applications this may include getting a response from the user. The value of the timer is between 1 and 10 minutes.

5.3.6 Multiple Dialogues

In the USSD GSM phase 2 specification only one dialogue between an MS and the network is allowed. If the MS receives a dialogue initiation in parallel to a currently ongoing dialogue, the new operation will be rejected with the “USSD Busy” error.

Once the dialogue is established between the MS and an end node in the GSM network another dialogue cannot be established in parallel. This means that a fix host that cannot be reached via the end node to which the dialogue is established, cannot be reached at all without first aborting the established dialogue, and then establish a new dialogue towards a different node; from which the terminal can be reached.

5.3.7 Addressing Aspects

USSD was designed for dialogues between the MS and a USSD application in the MSC, VLR or HLR. The MSISDN is transported in the dialogue part of the TCAP message. For example, when a mobile initiated dialogue is established toward an application in the HLR, the MSISDN and the HLR address is included.

For a mobile initiated dialogue the USSD application in the HLR is probably not the end application. The USSD application in the HLR will only work as a relay, and pass USSD operations between the GSM network and the external node.

5.3.8 Length of USSD String

According to the USSD GSM specification [GSM 02.90] the Invoke USSDRequest and the Invoke ProcessUSSDRequest can have USSD strings with a length of 160 octets. In addition, the length of the USSD string is restricted to the capabilities of the lower signalling layers (TCAP), which can be configured differently in different networks.

6. WAP and GSM USSD

6.1 Introduction

WAP requires a full duplex datagram service from the bearer network. Unfortunately, GSM USSD does not provide such a service. Instead, GSM USSD provides a two-way-alternate interactive service designed to convey short text strings between the mobile phone and a node in the GSM network. This chapter identifies the necessary parameter configurations and additional functionality that is needed in order to use GSM phase 2 USSD as a bearer of the WAP protocols. Note that other applications with a similar communication need as the WAP protocols (that is, a datagram service) may also benefit from using GSM phase 2 USSD as a bearer.

6.2 USSD Dialogue Control Protocol (UDCP)

The USSD dialogue provides a two-way-alternate interactive service to the user. This means that only the entity (mobile phone or network node) with the turn may send and its correspondent is permitted only to receive. To be able to use the USSD dialogue as a full duplex service a special protocol has to be specified that deals with the management of the dialogue. The protocol has to hide the two-way-alternate characteristics of the USSD dialogue to the upper layer, and allow the upper layer to use USSD as a full duplex service onto which datagrams can be sent and received.

The protocol designed to hide the complexity of the USSD dialogue is the USSD Dialogue Control Protocol (UDCP). UDCP is mapped directly onto the USSD protocol and is located in the mobile and the end node in the GSM network. The end points of UDCP are identical to the end points of the USSD dialogue.

The UDCP protocol is specified in chapter 7. The UDCP protocol is NOT part of the ETSI GSM Specifications.

6.3 Data Coding Scheme (DCS)

A USSD operation consists of the Data Coding Scheme (DCS) and the USSD String.

According to [GSM0290] all mobile originating operations the DCS MUST have the value 'Language unspecified' and 'Default alphabet'. According to [GSM0338] the DCS 'Language unspecified' and 'Default alphabet' should be coded as: 0x0F.

For network originated operations the DCS value is a matter for the network operator [GSM0290]. The network should ignore the value of the DCS [GSM0390]. WAP uses the DCS value 1110 xxxx. The least significant bits are used in the following way:

		Message Class:
xxxx	xx00	No message class.
xxxx	xx01	Class 1 Default meaning: ME-specific.
xxxx	xx10	Class 2 SIM specific message.
xxxx	xx11	Class 3 Default meaning: TE-specific.
		Message Coding:
xxxx	00xx	Reserved
xxxx	01xx	8 bit data
xxxx	10xx	Reserved
xxxx	11xx	Reserved

Class 1, Class 2 and Class 3 messages may be routed by the ME to user-defined destinations, but the user may override any default meaning and select their own routing.

6.4 Service Code (SC)

The service code identifies the USSD network-node and is an operator specific parameter; just like the SMS-Center address. The user has to manually enter the service code when invoking a service, or it could be entered once as a "Setting" in an application in the MS. When the service code is sent in the USSD string it has to have the format as defined in [GSM0290]; for example, in the string "*#SC* n number of characters #" SC is the service code.

The service code is only sent in the first operation of the dialogue and is there for routing purposes.

6.5 USSD Operation Timers

The Invoke USSDRequest timer will expire in the network if no Result USSDRequest message is received within the time set by the timer. As long as data is sent in the dialogue (USSD operations are continuously sent between the mobile and the network) this timer will not expire. During long idle periods the dialogue will be terminated by UDCP in order to free radio resources. This means that the Invoke USSDRequest timer can be ignored by UDCP.

The Invoke ProcessUSSDRequest timer limits the total length of a mobile initiated dialogue. When the timer expires the dialogue will be released regardless of whether data is sent in the dialogue or not. If this happens UDCP MAY try to re-establish the dialogue.

6.6 Multiple Dialogues

A mobile can have no more than one USSD dialogue established at any time. This means that once a USSD dialogue has been established between a mobile and the network, no more USSD dialogues can be established. However, datagrams sent over the USSD dialogue may belong to different applications. An application is identified by the port number in the datagram header.

Note that if datagrams from different applications are sent over the same USSD dialogue, they must all go through the same USSD network-node. This since a USSD dialogue can only be established between one mobile and exactly one USSD network-node. It is not possible to set up a second USSD dialogue from a different USSD network-node towards the same mobile.

6.7 Addressing Aspects

The service code is used by the mobile to address the USSD network-node towards which the USSD dialogue is established. The GSM network uses the service code to identify the USSD network-node. Formats for service codes are standardized by the ETSI and can be found in [GSM0290].

The IMSI is used to identify the mobile station. The mapping between the MSISDN and the IMSI is carried out in the operator's domain before the MSISDN is passed from the USSD node.

An external network node (WAP Gateway) may be connected to the USSD network-node. The address to the external node must be included in the USSD string. The address field must contain an address type (for example, IP-address, MSISDN, etc.) and the address itself. An address field for this purpose is included in the UDCP protocol header.

6.8 Length of the USSD String

In [GSM 0902] 160 octets is stated as the maximum length for the USSD string. Due to underlying signaling layers the maximum length of the USSD string depending on the message is:

Table 2 Length of USSD String

USSD operation	Max length [octets]
"Begin, Invoke ProcessUSSDRequest"	133
"End, Result ProcessUSSDRequest"	160
First "Continue, Invoke USSDRequest" in mobile initiated dialogue	154
"Begin, Invoke USSDRequest"	144
First "Continue, Result USSDRequest" in network initiated dialogue	154
Other messages	160

6.9 The USSD String

When USSD is used as a full duplex datagram transport mechanism, the same structure of the message that is used for the GSM short message (SMS) is used in the USSD string.

In GSM SMS, the User Data field may comprise just the short message itself or a Header in addition to the short message. The User Data Header (UDH) is defined in [GSM 0340]. The UDH consists of one or several Information Elements (IE). An Information Element has three fields: identifier, length and data. If a mobile receives an Information Element it does not recognize the Element will be discarded. The following table represents the generic structure of an IE:

Table 3 Structure of Information Element as defined in [GSM0340]

Field in Information Element	Description
Information Element Identifier (IEI) - 1 octet	Identifier
Information Element Length (IEL) - 1 octet	Length of IE Data
Information Element Data (IED) - 1 to n octets	IE Data

When the User Data Header is used for WAP over GSM USSD, the Information Element Identifiers are defined in [WINA].

6.9.1 UDCP Information Element

When UDCP is used an Information Element is included in the User Data Header. The Information Element contains the UDCP header.

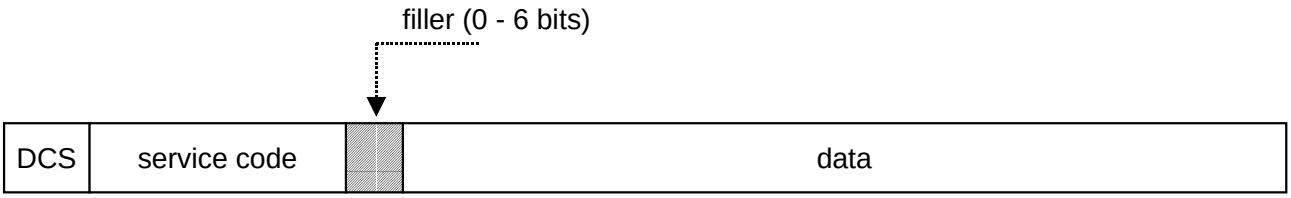
6.9.2 Encoding of the USSD String

Network Initiated USSD

A one-octet field called the Network Element Identifier (NEI) identifies the originating node in the network. The value of the NEI is configured by the operator.

Mobile Initiated USSD

The service code of the message is given a operator-defined value from the HPLMN-range that causes the message to be routed to the HLR. The message is then handed over to a WAP entity, either locally or in another network element (e.g. a WAP server).

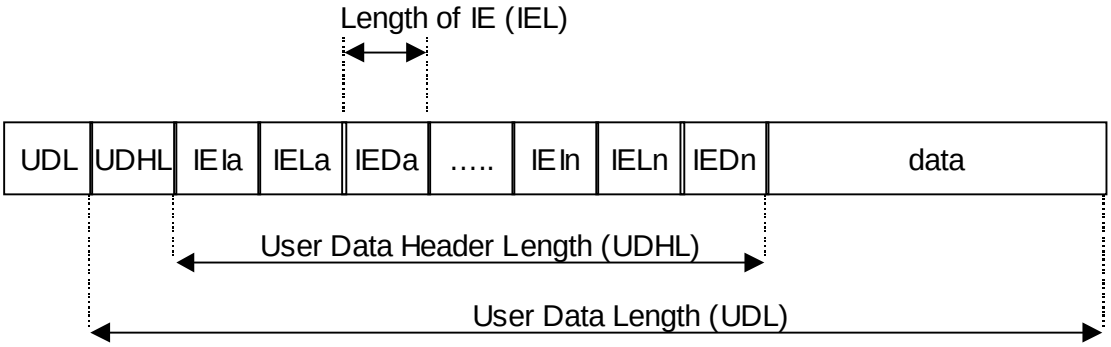


The service code, which is the only data the network is interested in, is encoded using the SMS default alphabet (7-bit characters), and it MUST comply to the GSM 02.90. The data starts on an octet boundary; filler bits are inserted between the service code and the data if needed.

The service code is only necessary in the first operation of the USSD dialogue, for subsequent operations the service code MUST be left out.

User Data Header

WAP has adopted the User Data Header from the SMS specification [GSM0340].



The UDH is always present when USSD is used as a transport mechanism for WAP.

7. USSD Dialogue Control Protocol

7.1 Goals and Requirements

The USSD Dialogue Control Protocol (UDCP) is defined in this chapter. The goals and requirements for the protocol are the following:

1. In the USSD string it must be possible to include an address to an external node. The address should indicate source/destination of the data in the USSD string. For example, address to a WAP Gateway.
2. The address in (1) can be of different types. IP address and MSISDN should be possible to use.
3. The USSD dialogue should remain established as long as there is data to be sent, or a timer expires. Terminating and establishing the dialogue between each data transmission should be avoided.
4. It should be possible to use the USSD dialogue as a full duplex service. The half duplex complexity should be hidden from the user.
5. It should be possible to re-establish the dialogue after failure.
6. The end points of UDCP should be identical to the end points of the USSD dialogue.

7.2 Architectural Overview

UDCP is located in the GSM mobile device and the USSD network-node in the GSM network. Although UDCP may convey the address to an external node (e.g. WAP Gateway) UDCP is terminated in the USSD network-node. This is illustrated below.

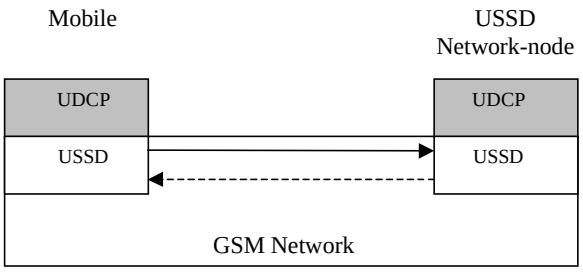


Figure 1 USSD and UDCP Architecture.

In the mobile the UDCP listens for incoming USSD dialogues. When a USSD dialogues is established to the mobile the UDCP provider locates the UDCP Information Element (IE) in the User Data Header and extracts the data portion. The data portion of the UDCP IE contains the UDCP PDU. If no UDCP IE can be found, the UDCP provider MUST ignore the USSD dialogue. Note that this doesn't mean that the USSD dialogue must be ignored by other applications in the phone.

Typically the UDCP user is WDP. The protocol includes the Port Number IE in the UDH. If segmentation and re-assembly is necessary, also the Fragmentation IE is included in the UDH. See [WDP].

7.3 Addressing

UDCP has been designed to handle two types of network architectures and addressing principles:

A. The Service Code of the USSD Dialogue is used to address both the USSD Node and the External Node.
B. The Service Code of the USSD Dialogue is used to address the USSD Node. An additional address is used to address the External Node.

Note: The addressing principle described in (A) is deprecated. An implementation MUST NOT initiate a dialogue using this addressing principle. However, an implementation MUST support dialogues initiated with this addressing principle for backward compatibility with WAP v1.1. In addition, an implementation SHOULD originate messages using the addressing principle described in (A), if the error code EXTADDRNOTSUPP is received that indicates that the other peer is a WAP v1.1 implementation only supporting the addressing principle described in (A).

In (A), once the dialogue has been established, the USSD Node function as a relay and passes data between the mobile and the External Node. The mobile can use the USSD dialogue to communicate only with the External Node identified by the Service Code. If the mobile wants to send data to another External Node, it first has to terminate the existing USSD dialogue and establish a new one.

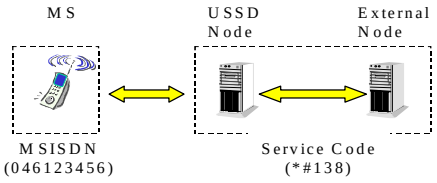


Figure 2 USSD Node and External Node both addressed by the same Service Code.

In the above figure the Service Code of the USSD Dialogue is mapped to the External Node. For example, *#138 may be mapped to the WAP Gateway. The Service Code is only sent in the first operation of the USSD Dialogue. Data in all subsequent operations of the USSD dialogue is passed to the External Node identified by *#138.

In (B), the address to the External Node is included in the UDCP header by using a special PDU for this purpose. The address to the External Node is sent in every USSD operation as part of the UDCP header. This makes it possible to use the same USSD dialogue for conversations with multiple External Nodes. The Service Code is only sent in the first USSD operation of a mobile initiated dialogue.

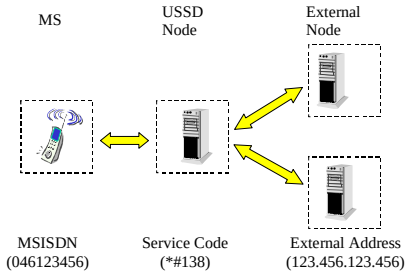


Figure 3 USSD Node addressed by Service Code and External Node addressed by an IP Address.

In the above figure the mobile may communicate with several External Nodes over the same USSD dialogue. For example, the first operation of the USSD dialogue may contain data destined for the

External Node with the IP address 123.456.123.456. The following operation may contain data destined for a different External Node with a different address.

7.4 UDCP Service Primitives

7.4.1 Introduction

This chapter defines the interface of UDCP to the next higher layer. The interface is defined using abstract service primitives.

7.4.2 UDCP-Data

This service is used to transfer data between the mobile and the USSD Node.

Primitive / Parameter	UDCP-Data			
	REQ	***IND***	***RES***	***CNF***
Service Code (mobile only)	M			
Address to External Node	M	M (=)		
Address to Mobile Phone	M	M(=)		
User Data	M	M(=)		

Service Code (mobile only)

The Service Code is sent in the first operation of the mobile initiated USSD dialogue. It is used by the GSM network to route the USSD operation to the correct USSD Node. Once the USSD Node has been located, a dialogue is established and the Service Code is no longer necessary.

Address to External Node

The address used to identify the External Node.

Address to Mobile Phone

The address used to identify the mobile phone.

User Data

The user data transported in the USSD string. This consists of two parts: Information Elements (IE) for the User Data Header and the message itself.

7.4.3 UDCP-Error

This service is used if there is an error. For example if the USSD service is not available from the network, or the USSD dialogue is terminated due to lack of radio coverage and can not be re-established.

Primitive / Parameter	UDCP-Error			
	REQ	***IND***	***RES***	***CNF***
Error Code		M		

Error Code

Error codes conveyed between UDCP providers are defined in the next chapter.

7.4.4 UDCP-Release

This service is used by the user to release the USSD dialogue.

Primitive	UDCP-Release			
Parameter	***REQ***	***IND***	***RES***	***CNF***
Release Code	O	C(=)		

Release Code

Release codes conveyed between UDCP providers are defined in the next chapter.

7.5 Data transfer

7.5.1 Motivation

Data transfer is the procedure to transfer data over the USSD dialogue between a mobile and a network-node in full duplex mode.

7.5.2 Procedure

The UDCP user sends data by issuing the UDCP-Data request primitive. The Data_Long PDU is used to transfer the address to the External Node between the mobile and the USSD Node. The Data PDU is only used for backward compatibility reasons. See separate chapter on Addressing.

If the UDCP user is WDP [WDP] the user data will consist of either the port number and the fragmentation Information Elements and the WDP user data or the WCMP Information Element.

The transmission mode of USSD dialogue is two-way alternate. During the time when the remote entity has the turn to send, the local UDCP user may initiate UDCP-Data request primitives. In that case, data has to be buffered up until the local UDCP provider has the turn to send.

If the UDCP provider has more data to send than it can send in one USSD operation, it should set the More To Send (MTS) flag in the Data_Long PDU. This indicates to the remote UDCP provider that it immediately should return an operation, in order to enable for the sender to send the rest of the data. When the local UDCP provider has no more data to send, it clears the MTS flag. When a UDCP provider receives a PDU with the MTS flag set, and it has no data to send, it MUST return the Receive Ready PDU. The Receive Ready PDU is a dummy PDU sent only to give the token to the remote entity.

7.6 Error handling

7.6.1 Motivation

If an error occurs that do not lead to a USSD dialogue release, the peer should be informed by returning the Error PDU with an appropriate error code.

If an error occurs that do lead to a USSD dialogue release the UDCP provider MAY re-establish the dialogue automatically.

7.6.2 Procedure

If an error occur that can not be mapped to any existing error code, the UDCP provider MUST return the Error PDU with the error code set to UNKNOWN.

If the UDCP provider receives a USSD string that it is unable to interpret it MUST return the Error PDU with the error code set to PROTOERROR.

If the UDCP provider receives a UDCP PDU with a version number different from 0x00, it MUST return the Error PDU with the error code set to UDCPVERSIONZERO to indicate that the supported version is 0x00.

The error code EXTADDRNOTSUPP is no longer used, but MUST be supported for backward compatibility reasons.

7.7 USSD Dialogue Release

7.7.1 Motivation

Before UDCP releases the USSD Dialogue it informs the remote UDCP provider.

7.7.2 Procedure

The dialogue release procedure may be triggered by different events. Depending on the event that triggered the release, different release codes are used.

The UDCP user has issued the UDCP-Release request primitive. Release code = USER.

The UDCP provider MAY monitor the USSD network timers in the network [GSM0902]. Before any of the timers expires the dialogue can be released and re-established. This will refresh the timers and the GSM network will not abort the dialogue due to a timer expiration. Release code = UTIMEOUT.

The UDCP provider MAY keep the dialogue established for some time after the last Data PDU is sent. This can be done by counting the number of subsequent RR PDUs. When the number of RR PDUs has reached a specified value, the dialogue is released. Release code = UIDLE.

The procedure to gracefully release the USSD dialogue is different depending on the type of USSD dialogue: mobile or network originated.

Mobile initiated dialogue

> If the mobile releases the dialogue it sends the Release Dialogue PDU in the Response USSDRequest operation. The network should respond by sending the Release Dialogue PDU in the Response ProcessUSSDRequest operation, and the dialogue is terminated. If the network releases the dialogue it send the Release Dialogue PDU in the Response ProcessUSSDRequest operation, and the dialogue is terminated.

Network initiated dialogue

> If the mobile releases the dialogue it sends the Release Dialogue PDU in the Response USSDRequest operation, and the dialogue is terminated. If the network releases the dialogue it send the Release Dialogue PDU in the Request USSDRequest operation. The mobile should respond by sending the Release Dialogue PDU in the Response USSDRequest operation, and the dialogue is terminated.

7.8 Timers and counters

7.8.1 Idle Timer (IT)

If a Receive Ready PDU is received and there is no data to be sent the receiver waits IT seconds before returning a response.

Recommended value : 2-10 seconds

7.8.2 Number Of RR (NumOfRR)

This counter counts the number of received subsequent RR PDUs. When the value reaches its maximum, MaxNumOfRR, the dialogue is released.

Recommended value of MaxNumOfRR : 1- 5

7.9 Protocol Data Units

7.9.1 General

The bits in an octet are ordered from left to right. The leftmost bit is bit 0 while the rightmost bit is bit 7. Bit 0 is the Most Significant Bit while bit 7 is the Least Significant Bit.

In a multi-octet field, the first octet is the most significant octet.

Reserved bits in the Protocol Data Units MUST be set to zero (0) by the sender.

7.9.2 Data PDU

The Data PDU is only used for backward compatibility reasons.

The Data PDU is used to send data between the Mobile and the USSD network-node. This is used when the USSD network-node can be identified by the Service Code (sent in the first USSD operation of the USSD dialogue).

Bit/Octet	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**
1	PDUtype = 0x00			RES	Version	MTS	RES	RES

7.9.3 Data_Long PDU

The Data_Long PDU is a special Data PDU used to send data between a Mobile and a node external to the GSM network. In this case the Service Code identifies the USSD network-node. In the USSD network-node the data is extracted and relayed to an external node. The address to the external node is included in the Data_Long PDU.

Bit/Octet	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**
1	PDUtype = 0x01			RES	Version	MTS	RES	RES
2	Address type			Address Length = N				
3 **…** **2+N**	Address Data							

7.9.4 Receive Ready (RR) PDU

This is a dummy PDU sent only to overcome the two-way alternate mode of the USSD dialogue.

Bit/Octet	0	1	2	3	4	5	6	7
1	PDUtype = 0x02			RES	Version	RES	RES	RES

7.9.5 Error PDU

The Error PDU is used to indicate an error to the peer.

Bit/Octet	0	1	2	3	4	5	6	7
1	PDUtype = 0x03			RES	Version	Error Reason		

7.9.6 Release Dialogue (RD) PDU

This is used by UDCP to indicate to the remote provider that the dialogue will be released.

Bit/Octet	0	1	2	3	4	5	6	7
1	PDUtype = 0x04			RES	Version	Release Code		

7.10 Header fields

7.10.1 More To Send flag (MTS)

The sender sets the More To Send flag to indicate that it has more data to send. The receiver must return an operation to enable the sender to send the rest of the data. If the receiver has no data to send it should send the Receive Ready PDU.

If the flag is clear it indicates that the sender has no data to send. This is used to poll the remote entity.

7.10.2 Address Field

In the network to mobile direction the Address field holds the source address and in the opposite direction it holds the destination address. This address corresponds to the destination/source address field in an SMS message, see [GSM 0340].

Encoding of the Address Type is specified in the following table:

Table 4 Address Types.

Address type	Assigned number
IPv4	0x00
IPv6	0x01
GSM_MSISDN	0x02
Reserved	0x03 to 0x07

Note: If the Address Type is GSM_MSISDN, the Address MUST be coded as defined in [GSM0340]. The semi-octet representation defined in [GSM0340] must be used. If the Address Type is IPv4 or IPv6, the address MUST be coded with the leftmost bit in the most significant octet as the most significant bit. See also [RFC791] and [RFC2373].

7.10.3 Version Flag

The current version number is 0x00.

7.10.4 Error Code

The following error codes have been defined:

Table 5 Error Codes.

Error	Code	Description
Unknown (UNKNOWN)	0x00	A generic error code indicating an unexpected error.
Protocol error (PROTOERR)	0x01	The received PDU could not be interpreted. The structure may be wrong.
UDCP Version Zero (UDCPVERSIONZERO)	0x02	Current version is 0.
External Addressing Not Supported (EXTADDRNOTSUPP)	0x03	The Data_Long PDU was received but the provider does not support addressing of external node. This Error Code is only used for backward compatibility reasons.

7.10.5 Release Code

The following release codes have been defined:

Table 6 Release Codes.

Error	Code	Description
Unknown (UNKNOWN)	0x00	A generic error release code.
USSD Timer Expiration (UTIMEOUT)	0x01	The dialogue is released by the UDCP provider to refresh a USSD network timer, that otherwise would terminate the dialogue at expiration.
Idle USSD Dialogue (UIDLE)	0x02	The dialogue is released by the UDCP provider since the dialogue has been idle for longer time than specified as the maximum idle time at the provider.
User abort (USER)	0x03	The UDCP user triggered the abort by issuing the UDCP-Release request primitive.

8. UDCP State Tables

8.1 Event Processing

The interface to the next higher layer is defined by the UDCP service primitives. When the UDCP user issues a primitive the corresponding event is generated. The next lower layer is the USSD service of the GSM phase 2 network. The following GSM phase 2 USSD operations are used by the UDCP state tables:

Table 7 GSM phase 2 operation used in the State Tables

GSM phase 2 USSD Operation	Description
USSDRequest	Initiates a dialogue from the network. Used to invoke operations from the network within an established dialogue.
ProcessUSSDRequest	Initiates a dialogue from the mobile

For a complete description of how the USSD operations are used and how the USSD dialogue is initiated and terminated, see [GSM0290], [GSM0390] and [GSM0490]. When a USSD operation has been received the USSD String is extracted, the UDCP control information analyzed and an event is generated depending on the PDU type. For example, *RcvData* means that a Data PDU (or Data_Long PDU) has been received.

An event is validated before it is processed. The following tests are performed, and if no action is taken, the event is processed according to the state tables.

Table 8 Test of Events

Test	Action
UDCP-Data.req and size of UserData > max size of USSD string	Generate UDCP-Error.ind
Illegal PDU type or erroneous structure	Send Error (PROTOERR)
Received PDU with version != 0x00	Send Error (UDCPVERSIONZERO)
Received Data_Long PDU without support for addressing to external node	Send Error (EXTADDRNOTSUPP)
Any other event not handled by the state tables	Ignore

The Error Code EXTADDRNOTSUPP is only used for backward compatibility reasons.

8.2 Actions

8.2.1 Timers

The following timer actions can be used in the state tables:

Start timer, <value>
Starts the timer with the specified interval value. If the timer is already running, it is re-started with the new value.

Stop timer
Stop the timer without generating an event.

8.2.2 Counters

The following counter actions can be used in the state tables:

<counter>++
Add one to the counter value.

8.2.3 Messages

The following message actions can be used in the state tables:

USSDRequest
At the mobile this action sends the Invoke ProcessUSSDRequest. In the network this action sends the Invoke USSDRequest.

USSDResponse
At the mobile this action sends the Result USSDRequest. In the network this action sends the Result ProcessUSSDRequest.

8.2.4 Output Buffer

The UDCP provider manages an output buffer to queue outgoing messages. The buffer has the following methods:

Empty()
Returns True if the buffer is empty. Returns False if the buffer is not empty.

Length()
Returns the number of messages in the buffer. The maximum number of messages in the queue is MAX_BUF.

Queue()
Adds another message to the queue.

Dequeue()
De-queues a message. The message is removed from the queue.

8.2.5 USSD Dialogue

The following USSD Dialogue actions can be used in the state tables:

Release Dialogue
Release the USSD dialogue. How this action is executed is implementation dependent.

Initiate Dialogue
Initiate the USSD dialogue. How this action is executed is implementation dependent.

8.2.6 States

The following states are used in the state tables.

IDLE
In this state there is no USSD dialogue established.

LISTEN
In this state there is no USSD dialogue established. The provider is listening for an incoming USSD dialogue invocation.

WAIT NETWORK

The UDCP provider in the mobile is waiting for a USSD operation to be sent from the network.

WAIT MOBILE

The UDCP provider in the network is waiting for a USSD operation to the sent from the mobile.

WAIT USER

The UDCP provider is waiting for the local user to issue a service primitive. The Idle Timer is running.

8.3 Mobile Initiated USSD Dialogue

8.3.1 Mobile

UDCP Mobile IDLE Mobile Initiated Dialogue			
Event	**Condition**	**Action**	**Next State**
UDCP-Data.req		Initiate Dialogue Send USSDRequest (Data PDU) ReleaseDlg = False NumOfRR = 0	WAIT NETWORK

UDCP Mobile WAIT NETWORK Mobile Initiated Dialogue			
Event	**Condition**	**Action**	**Next State**
RcvData	MTS == True	Generate UDCP-Data.ind Send USSDResponse (RR PDU) NumOfRR = 0	WAIT NETWORK
	MTS == False	Generate UDCP-Data.ind Start timer, IT NumOfRR = 0	WAIT USER
	!OutBuf.Empty()	Generate UDCP-Data.ind SendPDU = OutBuf.Dequeue() SendPDU.MTS = OutBuf.Empty() Send USSDResponse (SendPDU) NumOfRR = 0	WAIT NETWORK
	ReleaseDlg == True	Send USSDResponse (RD PDU) NumOfRR = 0	WAIT NETWORK
RcvRR	!OutBuf.Empty()	SendPDU = OutBuf.Dequeue() SendPDU.MTS = OutBuf.Empty() Send USSDResponse (SendPDU) NumOfRR = 0	WAIT NETWORK
	NumOfRR < MaxNumOfRR	Start timer, IT NumOfRR++	WAIT USER
	NumOfRR == MaxNumOfRR	Send USSDResponse (RD PDU)	WAIT NETWORK
	ReleaseDlg == True	Send USSDResponse (RD PDU)	WAIT NETWORK
RcvRD		Release Dialogue	IDLE
UDCP-Data.req	OutBuf.Length() < MAX_BUF	OutBuf.Queue(Data PDU) NumOfRR = 0	WAIT NETWORK
	OutBuf.Length() == MAX_BUF	Generate UDCP-Error.ind (BUFFEROVERFLOW) NumOfRR = 0	
UDCP-Release.req		ReleaseDlg = True	

UDCP Mobile WAIT USER Mobile Initiated Dialogue			
Event	**Condition**	**Action**	**Next State**
UDCP-Data.req		Send USSDResponse (Data PDU)	WAIT NETWORK
TimerTO		Send USSDResponse (RR PDU)	
UDCP-Release.req		Send USSDResponse (RD PDU)	

8.3.2 Network

UDCP Network LISTEN Mobile Initiated Dialogue			
Event	**Condition**	**Action**	**Next State**
RcvData	MTS == True	Generate UDCP-Data.ind Send USSDRequest (RR PDU) ReleaseDlg = False NumOfRR = 0	WAIT MOBILE
	MTS == False	Generate UDCP-Data.ind Start timer, IT ReleaseDlg = False NumOfRR = 0	WAIT USER

UDCP Network WAIT MOBILE Mobile Initiated Dialogue			
Event	**Condition**	**Action**	**Next State**
RcvData	MTS == True	Generate UDCP-Data.ind Send USSDRequest (RR PDU) NumOfRR = 0	WAIT MOBILE
	MTS == False	Generate UDCP-Data.ind Start timer, IT NumOfRR = 0	WAIT USER
	!OutBuf.Empty()	Generate UDCP-Data.ind SendPDU = OutBuf.Dequeue() SendPDU.MTS = OutBuf.Empty() Send USSDRequest (SendPDU) NumOfRR = 0	WAIT MOBILE
	ReleaseDlg == True	Send USSDResponse (RD PDU) NumOfRR = 0	LISTEN
RcvRD		Send USSDResponse (RD PDU)	LISTEN
RcvRR	!OutBuf.Empty()	SendPDU = OutBuf.Dequeue() SendPDU.MTS = OutBuf.Empty() Send USSDRequest (SendPDU) NumOfRR = 0	WAIT MOBILE
	NumOfRR < MaxNumOfRR	Start timer, IT NumOfRR++	WAIT USER
	NumOfRR == MaxNumOfRR	Send USSDResponse (RD PDU) Release Dialogue	LISTEN
	ReleaseDlg == True	Send USSDResponse (RD PDU) Release Dialogue	LISTEN
UDCP-Data.req	OutBuf.Length() < MAX_BUF	OutBuf.Queue(Data PDU) NumOfRR = 0	WAIT MOBILE
	OutBuf.Length() == MAX_BUF	Generate UDCP-Error.ind (BUFFEROVERFLOW) NumOfRR = 0	
UDCP-Release.req		ReleaseDlg = True	WAIT MOBILE

UDCP Network WAIT USER Mobile Initiated Dialogue			
Event	**Condition**	**Action**	**Next State**
UDCP-Data.req		Send USSDRequest (Data PDU)	WAIT MOBILE
TimerTO		Send USSDRequest (RR PDU)	
UDCP-Release.req		Send USSDResponse (RD PDU) Release Dialogue	LISTEN

8.4 Network Initiated USSD Dialogue

8.4.1 Mobile

UDCP Mobile LISTEN Network Initiated Dialogue			
Event	**Condition**	**Action**	**Next State**
RcvData	MTS == True	Generate UDCP-Data.ind Send USSDResponse (RR PDU) ReleaseDlg = False NumOfRR = 0	WAIT NETWORK
	MTS == False	Generate UDCP-Data.ind Start timer, IT ReleaseDlg = False NumOfRR = 0	WAIT USER

UDCP Mobile WAIT NETWORK Network Initiated Dialogue			
Event	**Condition**	**Action**	**Next State**
RcvData	MTS == True	Generate UDCP-Data.ind Send USSDResponse (RR PDU) NumOfRR = 0	WAIT NETWORK
	MTS == False	Generate UDCP-Data.ind Start timer, IT NumOfRR = 0	WAIT USER
	!OutBuf.Empty()	Generate UDCP-Data.ind SendPDU = OutBuf.Dequeue() SendPDU.MTS = OutBuf.Empty() Send USSDResponse (SendPDU) NumOfRR = 0	WAIT NETWORK
	ReleaseDlg == True	Send USSDResponse (RD PDU) Release Dialogue	LISTEN
UDCP-Data.req	OutBuf.Length() < MAX_BUF	OutBuf.Queue(Data PDU) NumOfRR = 0	WAIT NETWORK
	OutBuf.Length() == MAX_BUF	Generate UDCP-Error.ind (BUFFEROVERFLOW) NumOfRR = 0	
RcvRR	!OutBuf.Empty()	SendPDU = OutBuf.Dequeue() SendPDU.MTS = OutBuf.Empty() Send USSDResponse (SendPDU) NumOfRR = 0	WAIT NETWORK
	NumOfRR < MaxNumOfRR	Start timer, IT NumOfRR++	WAIT USER
	NumOfRR == MaxNumOfRR	Send USSDResponse (RD PDU) Release Dialogue	LISTEN
	ReleaseDlg == True	Send USSDResponse (RD PDU) Release Dialogue	LISTEN
RcvRD		Send USSDResponse (RD PDU) Release Dialogue	LISTEN
UDCP-Release.req		ReleaseDlg = True	WAIT NETWORK

UDCP Mobile WAIT USER Network Initiated Dialogue			
Event	**Condition**	**Action**	**Next State**
UDCP-Data.req		Send USSDResponse (Data PDU)	WAIT NETWORK
TimerTO		Send USSDResponse (RR PDU)	
UDCP-Release.req		Send USSDResponse (RD PDU) Release Dialogue	LISTEN

8.4.2 Network

UDCP Network IDLE Network Initiated Dialogue			
Event	**Condition**	**Action**	**Next State**
UDCP-Data.req		Initiate Dialogue Send USSDRequest (Data PDU) ReleaseDlg = False NumOfRR = 0	WAIT MOBILE

UDCP Network WAIT MOBILE Network Initiated Dialogue			
Event	**Condition**	**Action**	**Next State**
RcvData	MTS == True	Generate UDCP-Data.ind Send USSDRequest (RR PDU) NumOfRR = 0	WAIT MOBILE
	MTS == False	Generate UDCP-Data.ind Start timer, IT NumOfRR = 0	WAIT USER
	!OutBuf.Empty()	Generate UDCP-Data.ind SendPDU = OutBuf.Dequeue() SendPDU.MTS = OutBuf.Empty() Send USSDRequest (SendPDU) NumOfRR = 0	WAIT MOBILE
	ReleaseDlg == True	Send USSDRequest (RD PDU) NumOfRR = 0	WAIT MOBILE
RcvRR	!OutBuf.Empty()	SendPDU = OutBuf.Dequeue() SendPDU.MTS = OutBuf.Empty() Send USSDResponse (SendPDU) NumOfRR = 0	WAIT MOBILE
	NumOfRR < MaxNumOfRR	Start timer, IT NumOfRR++	WAIT USER
	NumOfRR == MaxNumOfRR	Send USSDRequest (RD PDU)	WAIT MOBILE
	ReleaseDlg == True	Send USSDRequest (RD PDU)	WAIT MOBILE
UDCP-Data.req	OutBuf.Length() < MAX_BUF	OutBuf.Queue(Data PDU) NumOfRR = 0	WAIT MOBILE
	OutBuf.Length() == MAX_BUF	Generate UDCP-Error.ind (BUFFEROVERFLOW) NumOfRR = 0	
UDCP-Release.req		ReleaseDlg = True	WAIT MOBILE
RcvRD		Release Dialogue	IDLE

UDCP Network WAIT USER Network Initiated Dialogue			
Event	**Condition**	**Action**	**Next State**
UDCP-Data.req		Send USSDRequest (Data PDU)	WAIT MOBILE
IdleTimerTO		Send USSDRequest (RR PDU)	
UDCP-Release.req		Send USSDRequest (RD PDU)	

8.5 Example of WDP and UDCP

The following example illustrates how the WDP protocol inter works with the UDCP protocol. Two datagrams, WDP SDU(1) and WDP SDU(2), are sent from the mobile to the network.

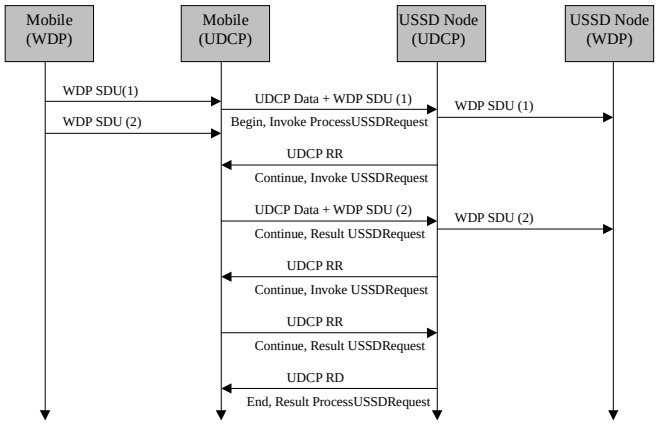


Figure 4 Example of WDP and UDCP inter working.

The WDP provider starts off by sending two subsequent datagrams to the UDCP provider. The UDCP provider establishes a USSD dialogue to the USSD Node identified by the Service Code. The Service Code is provided to the UDCP provider from WDP as a parameter of the service primitive. In the first operation the MTS flag is set(not indicated in the figure) in the Data PDU. When the USSD Node receives the PDU with the MTS flag it immediately returns the dummy RR PDU to allow for the mobile to send another USSD operation. Recall that the USSD dialogue is half-duplex: the mobile can not send the next operation before the network has returned an operation. Once the dummy RR PDU has been received by the mobile, the next Data PDU is sent with the last WDP SDU. This time the MTS is not set, since the mobile has no more data to send. When the USSD Node receives the Data PDU with the MTS flag clear it waits for a period of time specified by the Idle Timer, after that it sends the RR PDU to poll the remote entity. After having received MaxNumOfRR number of RR PDUs the USSD Node sends the RD PDU to release the dialogue. The value of MaxNumOfRR is an implementation issue. In this example MaxNumOfRR =1.

Appendix A. Static Conformance Requirement

This static conformance requirement defines a minimum set of features that can be implemented to ensure that the implementation will be able to inter-operate. A feature can be optional or mandatory. If a UDCP implementation does not support an optional feature, transmission should occur without error, but may not be optimal.

A.1. Functions

Identifier	Structure	Reference	Status
UDCP-001	Addressing using service code.	7.5	M
UDCP-002	Addressing using service code and an external address.	7.5	M
UDCP-003	Error handling.	7.6	M
UDCP-004	Dialogue release issued by user.	7.7	M
UDCP-005	Dialogue release to refresh USSD network timer.	7.7	O
UDCP-006	Dialogue release due to an idle dialogue.	7.7	O

Note: The addressing principle described by UDCP-001 is deprecated. An implementation MUST NOT initiate a dialogue using this addressing principle. However, an implementation MUST support dialogues initiated with this addressing principle for backward compatibility with WAP v1.1. In addition, an implementation SHOULD originate messages using the addressing principle described by UDCP-001, if the error code EXTADDRNOTSUPP is received that indicates that the other peer is a WAP v1.1 implementation only supporting the addressing principle described by UDCP-001.

Appendix B. History and Contact Information

Document history		
Date	**Status**	**Comment**
30-April-1998	Specification	First version.
07-May-1999	Specification	Copyright updated.
31-May-1999	Specification	The corrigendum WPG-UDCP-2 incorporated into the document: • Removed - Appendix A. PICS Proforma
15-July-1999	Specification	Third version.
19-Feb-2000	Proposed	Incorporated the following change requests: CR-Ericsson-UDCP-19-Nov-1999, CR-Ericsson-UDCP-27-Jan-2000 and CR-Ericsson-UDCP-07-Feb-2000
23-May-2000	Approved	Incorporated the SCD WAP-204_001-WAPOverGSMUSSD (originally named WAP-103_001-WAPOverGSMUSSD)
Contact Information http://www.wapforum.org. technical-comments@wapforum.org		