

WAP WTAI (PDC)

Version 30-Apr-1998

Wireless Application Protocol Wireless Telephony Application Interface Specification

PDC Specific Addendum

Disclaimer:

This document is subject to change without notice.

Contents

1	SCOPE.....	3
2	DOCUMENT STATUS.....	4
2.1	COPYRIGHT NOTICE.....	4
2.2	ERRATA.....	4
2.3	COMMENTS	4
3	REFERENCES	5
3.1	NORMATIVE REFERENCES.....	5
4	DEFINITIONS AND ABBREVIATIONS.....	6
4.1	DEFINITIONS.....	6
4.2	ABBREVIATIONS	6
5	PDC SPECIFIC LIBRARY.....	7
5.1	NETWORK FUNCTIONS.....	7
5.2	CALL REJECT.....	7
5.3	CALL HOLD	8
5.4	CALL TRANSFER.....	9
5.5	JOIN MULTIPARTY	10
5.6	RETRIEVE FROM MULTIPARTY.....	11
	APPENDIX A. WTA URI AND WMLSCRIPT FUNCTION LIBRARIES.....	12

1 Scope

Wireless Application Protocol (WAP) is a result of continuous work to define an industry wide specification for developing applications that operate over wireless communication networks. The scope for the WAP Forum is to define a set of specifications to be used by service applications. The wireless market is growing very quickly, and reaching new customers and services. To enable operators and manufacturers to meet the challenges in advanced services, differentiation and fast/flexible service creation WAP defines a set of protocols in transport, session and application layers. For additional information on the WAP architecture, refer to "*Wireless Application Protocol Architecture Specification*" [WAP].

This document is an addendum to the *Wireless Telephony Application Interface* (WTAI). While WTAI defines an API that is valid for all supported types of mobile networks, this document outlines functions that are specific to PDC networks.

2 Document Status

This document is available online in the following formats:

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

2.1 Copyright Notice

© Copyright Wireless Application Protocol Forum Ltd, 1998 all rights reserved.

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

The following section describes references relevant to this document.

3.1 Normative references

- [RFC2119] "Key words for use in RFCs to Indicate Requirement Levels", S. Bradner, March 1997. URL: <ftp://ds.internic.net/rfc/rfc2119.txt>
- [RFC1630] "Uniform Resource Identifiers (URI)", T. Berners-Lee, et al., June 1994. URL: <ftp://ds.internic.net/rfc/rfc1630.txt>
- [WAP] "Wireless Application Protocol Architecture Specification, version 0.9", WAP Forum, 1997. URL: <http://www.wapforum.org/>
- [WMLScript] "WMLScript Language Specification", WAP Forum, 1998. URL: <http://www.wapforum.org/>
- [WTA] "Wireless Telephony Application Specification", WAP Forum, 1998. URL: <http://www.wapforum.org/>
- [WTAI] "Wireless Telephony Application Interface Specification", WAP Forum, 1997. URL: <http://www.wapforum.org/>

4 Definitions and abbreviations

The following section describes definitions and abbreviations common to this document.

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY" and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

4.1 Definitions

The following are terms and conventions used throughout this specification.

WMLScript - a scripting language used to program the mobile device. WMLScript is an extended subset of the JavaScript™ scripting language.

4.2 Abbreviations

For the purposes of this specification, the following abbreviations apply.

API	Application Programming Interface
GSM	Global System for Mobile Communication
PDC	Pacific Digital Cellular System
RFC	Request For Comments
URI	Uniform Resource Identifier [RFC1630]
WAP	Wireless Application Protocol [WAP]
WTA	Wireless Telephony Applications [WTA]
WTAI	Wireless Telephony Applications Interface [WTAI]

5 PDC Specific Library

In addition to the WTA functions defined in [WTAI], PDC networks also supports the functions specified in this chapter.

5.1 Network Functions

The functions defined in this chapter follows the same function definition format as the one used in [WTAI]. Technical terms used in this chapter, e.g. events and error codes, are also explained in [WTAI].

Name:	WTAPDC
Library ID:	520
Description:	This library contains functions that are unique to PDC networks.

5.2 Call Reject

Description	
Rejects an unanswered call.	
URI:	wtai://pdc/cr;<id> [! <result>]
WMLScript:	reject(id)
Function ID:	0
Parameters:	<id> = String: The identity of the call to be rejected.
Output:	<result> = String: The return value is the identity of the rejected call or a negative number in case of failure, the WTAI error code.
Examples:	URI: wtai://pdc/cr; 1 WMLScript: WTAPDC.reject ("1");
Associated Events:	-
Notes: -	

5.3 Call Hold

Description	
Puts an answered call on hold.	
URI:	wtai://pdc/ch;<id> [! <result>]
WMLScript:	hold(id);
Function ID:	1
Parameters:	<id> = String: The identity of the call to be put on hold.
Output:	<result> = String: The return value is the identity of the held call or a negative number in case of failure, the WTAI error code.
Examples:	URI: wtai://pdc/ch; 1 WMLScript: WTAPDC.hold ("1");
Associated Events:	-
Notes: -	

5.4 Call Transfer

Description	
Transfers an unanswered call to another party.	
URI:	wtai://pdc/ct; <id> ; <dest> [! <result>]
WMLScript:	transfer(id);
Function ID:	2
Parameters:	<p><id> = String: The identity of the call to be transferred.</p> <p><dest> = String: The destination to where the call should be transferred (any valid phone number).</p>
Output:	<p><result> = String: The return value is the identity of the transferred call or a negative number in case of failure, the WTAI error code.</p>
Examples:	<p>URI: wtai://pdc/ct; 1;" +15551234"</p> <p>WMLScript: WTAPDC. transfer ("1" ," +15551234");</p>
Associated Events:	-
Notes:	-

5.5 Join Multiparty

Description	
<p>This function is partly used for establishing a multiparty call, and partly for joining new parties to an existing multiparty.</p> <p>Establish a multiparty: Joins an active call with a call on hold. A multiparty call (with a unique “id”) is established.</p> <p>Add new party: Joins an active call with a multiparty on hold.</p> <p>How a call is put on hold is described in [WTAI].</p>	
URI:	wtai://pdc/jm [! <result>]
WMLScript:	Multiparty;
Function ID:	3
Parameters:	-
Output:	<p><result> = String:</p> <p>The return value is the identity of the multiparty call or a negative number in case of failure, the WTAI error code.</p>
Examples:	<p>URI: wtai://pdc/jm</p> <p>WMLScript: WTAPDC.multiparty;</p>
Associated Events:	-
Notes: -	

5.6 Retrieve from Multiparty

Description	
Separates a certain party from a multiparty call for a private conversation. The rest of the multiparty is put on hold.	
URI:	wtai://pdc/rm;<id> [! <result>]
WMLScript:	Retrieve("1");
Function ID:	4
Parameters:	<id> = String: The identity of the call to be retrieved from the multiparty.
Output:	<result> = String: The return value is the identity of the retrieved call or in case of failure a negative number, the WTAI error code.
Examples:	URI: wtai://pdc/rm;1 WMLScript: WTAPDC.retrieve ("1");
Associated Events:	-
Notes: -	

Appendix A. WTA URI and WMLScript Function Libraries

In the table below, the URI and WMLScript Function Libraries Calls valid for PDC networks are summarised. The arguments have been left out in order to increase readability. The figures in the column named "Lib/Func ID" denote the *Library* and *Function IDs*.

Table 1 , URI's and WMLScript Functions

<i>Lib/Func ID</i>	<i>URI</i>	<i>WMLScript call</i>	<i>Description</i>
520.0	wtai://pdc/cr	WTAPDC.reject	Reject an incoming call
520.1	wtai://pdc/ch	WTAPDC.hold	Put a call on hold
520.2	wtai://pdc/ct	WTAPDC.transfer	Transfer an unanswered call
520.3	wtai://pdc/jm	WTAPDC.multiparty	Join/create a multiparty call
520.4	wtai://pdc/rm	WTAPDC.retrieve	Retrieves a party from a multiparty call