Wirless Telephony Application Interface

ANSI136-specific Addendum

Version 08-Sep-2001

Wireless Application Protocol
WAP-269-WTAIIS136-20010908

This document is available online in PDF format at http://www.wapforum.org/.

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

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

Document History	
WAP-269-WTAIIS136-20010710-p	Proposed
WAP-269-WTAIIS136-20010908	Current

Contents

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"* [WAPARCH].

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 ANSI 136 networks.

2. References

2.1. Normative References

[CREQ]   WAP-221, "Specification of WAP Conformance Requirements", WAP Forum™, 25-Apr-2001
                                   URL: http://www.wapforum.org/

[RFC2119]   "Key words for use in RFCs to Indicate Requirement Levels". S. Bradner. March 1997
                                   URL: http://www.ietf.org/rfc/rfc2119.txt

[WMLScript]   WAP-193, "WMLScript Language Specification", WAP Forum™, 25-Oct-2000 URL:
                                   http://www.wapforum.org/

[WTA]   WAP-266, "Wireless Telephony Application Specification", WAP Forum™, 08-Sep-2001 URL:
                                   http://www.wapforum.org/

[WTAI]   WAP-268, "Wireless Telephony Application Interface Specification", WAP Forum™,
                                   08-Sep-2001 URL: http://www.wapforum.org/

2.2. Informative References

[WAPARCH]   WAP-210, "WAP Architecture", WAP Forum™, 12-Jul-2001 URL: http://www.wapforum.org/

3. Terminology and Conventions

3.1. Conventions

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].

All sections and appendixes, except "Scope" and "Introduction", are normative, unless they are explicitly indicated to be informative.

3.2. Definitions

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

3.3. Abbreviations

API   Application Programming Interface
ANSI136   TDMA Cellular/PCS – Radio Interface – Mobile Station – Base Station Compatibility Standard
RFC   Request For Comments
WAP   Wireless Application Protocol
WTA   Wireless Telephony Applications
WTAI   Wireless Telephony Applications Interface

4. Introduction

The WAP WTAI features provide the means to create Telephony Applications, using a WTA user-agent with the appropriate WTAI function libraries. A typical example is to set-up a mobile originated call using the WTAI functions accessible from either a WML deck/card or WMLScript. The application model for WTA is described in [WTA].

The ANSI 136 addendum extends the support of [WTAI] to ANSI136 technology devices by specifying additional events and functions.

5. Network Specific WTAI – ANSI136

In addition to the WTAI functions defined in [WTAI], ANSI 136 networks also support the functions specified in this chapter. An ANSI136 WTA implementation MUST support the Network Common WTAI – Voice Call model specified in [WTAI].

5.1. WTA Events

These events are related to ANSI 136 devices. All WTA event parameters are conveyed as strings. An ANSI136 WTA implementation MUST support the Network Specific WTAI – ANSI136 events specified in this chapter.

5.1.1. wtaev-ansi136/ia

Event Name:	IncomingAlert
Event ID:	wtaev-ansi136/ia
Parameters:	*callHandle*, *alertSequence*
Description:	Indicates an incoming alert has been received.

The *callHandle* parameter contains the call handle for the voice call that received the alert. (See [WTAI] for a description of the call handle.)

The *alertSequence* parameter contains the alert sequence information.

5.1.2. wtaev-ansi136/if

Event Name:	IncomingFlash
Event ID:	wtaev-ansi136/if
Parameters:	*callHandle*, *flashSequence*
Description:	Indicates an incoming flash has been received.

The *callHandle* parameter contains the call handle for the voice call that received the flash. (See [WTAI] for a description of the call handle.)

The *flashSequence* parameter contains the flash sequence information.

5.2. WMLScript Functions

The functions defined in this chapter follow the same function definition format as the one used in [WTAI]. Technical terms used in this chapter, eg events and error codes, are also explained in [WTAI]. An ANSI136 WTA implementation MUST support the Network Specific WTAI – ANSI136 functions specified in this chapter.

Name:	WTAANSI136
Library ID:	517
Description:	This library contains functions that are available on ANSI 136 implementations of WTA.

5.2.1. WTAANSI136.sendFlash

Function:	sendFlash(*callHandle*,*flashSequence*)
Function ID:	0
Description:	Sends a flash code sequence through an active voice call.

The *callHandle* parameter identifies the voice call on which to send the flash code sequence. (See [WTAI] for a description of the call handle.)

The *flashSequence* parameter contains the flash code sequence to send.

This function returns an empty `string` if successful, or returns `invalid` if the function fails.

Permission Types:	BLANKET, CONTEXT, SINGLE (see [WTA]).
Parameters:	*callHandle* = handle
	flashSequence = string
Return value:	empty `string` or `invalid`
Associated Events:	-
Exceptions:	If the *callHandle* parameter does not refer to an existing ANSI 136 voice call through which a flash code sequence can be sent, this function returns `invalid`.
Example:	var flag = WTAANSI136.sendFlash(handle, "123");

5.2.2. WTAANSI136.sendAlert

Function:	sendAlert(*callHandle*,*alertSequence*)
Function ID:	1
Description:	Sends an alert code sequence through an active voice call.
	The *callHandle* parameter identifies the voice call on which to send the alert code sequence. (See [WTAI] for a description of the call handle.)
	The *alertSequence* parameter contains the alert code sequence to send.
	This function returns an empty `string` if successful, or returns `invalid` if the function fails.
Permission Types:	BLANKET, CONTEXT, SINGLE (see [WTA]).
Parameters:	*callHandle* = handle
	alertSequence = string
Return value:	empty `string` or `invalid`
Associated Events:	-
Exceptions:	If the *callHandle* parameter does not refer to an existing ANSI 136 voice call through which an alert code sequence can be sent, this function returns `invalid`.
Example:	var flag = WTAANSI136.sendAlert(handle, "123");

Appendix A.   Static Conformance Requirements   (Normative)

The notation used in this appendix is specified in [CREQ].

A 1 Client features

A 1.1 WTA Events

Item	Function	Reference	Status	Requirement
WTAIANSI136-E-C-001	IncomingAlert (wtaev-ansi136/ia)	5.1.1	M	
WTAIANSI136-E-C-002	IncomingFlash (wtaev-ansi136/if)	5.1.2	M	

A 1.2 WMLScript Functions

Item	Function	Reference	Status	Requirement
WTAIANSI136-S-C-001	WTAANSI136.sendFlash	5.2.1	M	
WTAIANSI136-S-C-002	WTAANSI136.sendAlert	5.2.2	M	

A 1.3 WMLScript Bytecode Interpeter Capabilities

Item	Function	Reference	Status	Requirement
WTAIANSI136-INT-C-001	Supports ANSI 136 Network WTAI library identifier	5.2	M	WMLS:MCF
WTAIANSI136-INT-C-002	Supports ANSI 136 Network WTAI function identifiers	5.2	M	WMLS:MCF

A 2 Server features

A 2.1 WMLScript Encoder Capabilities

Item	Function	Reference	Status	Requirement
WTAIANSI136-ENC-S-001	Supports ANSI 136 Network WTAI library identifier	5.2	M	WMLS:MSF
WTAIANSI136-ENC-S-002	Supports ANSI 136 Network WTAI function identifiers	5.2	M	WMLS:MSF

Appendix B.   WMLScript Function Libraries   (Informative)

In the table below, the WMLScript Function Libraries Calls valid for ANSI 136 networks are summarised. The arguments have been left out in order to increase readability. The values in the column named "Lib/Func ID" denote the *Library* and *Function ID*s.

Lib/Func ID	WMLScript call	Description
517.0	WTAANSI136.sendFlash	Send a flash code
517.1	WTAANSI136.sendAlert	Send an alert code

Table 1 , WMLScript Functions

Appendix C.   Change History   (Informative)

Type of Change	Date	Section	Description
Class 0	08-Sep-2001		The initial version of this document.