

WAP WTAI (ANSI 136)

WAP-172-WTAIIS136

Version 07-Jul-2000

Wireless Application Protocol Wireless Telephony Application Interface Specification

ANSI 136 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	ANSI 136 SPECIFIC LIBRARY	7
5.1	WTA EVENTS	7
5.1.1	<i>wtaev-ansi136/ia</i>	7
5.1.2	<i>wtaev-ansi136/if</i>	7
5.2	WMLSCRIPT FUNCTIONS	7
5.2.1	<i>WTAANSI136.sendFlash</i>	7
5.2.2	<i>WTAANSI136.sendAlert</i>	8
	APPENDIX A WMLSCRIPT FUNCTION LIBRARIES.....	9
	APPENDIX B STATIC CONFORMANCE REQUIREMENTS	10
B.1	CLIENT FEATURES.....	10
B.2	SERVER FEATURES.....	10
	APPENDIX C SPECIFICATION-TRACK DOCUMENT HISTORY	11

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 ANSI 136 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, 2000. Terms and conditions of use are available from the Wireless Application Protocol Forum Ltd. web site at <http://www.wapforum.org/docs/copyright.htm>.

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

- [RFC2396] "Uniform Resource Identifiers (URI): Generic Syntax." T. Berners-Lee, R. Fielding, L. Masinter, et al., August 1998. URL: <http://www.ietf.org/rfc/rfc2396.txt>
- [RFC2119] "Key words for use in RFCs to Indicate Requirement Levels", S. Bradner, March 1997. URL: <http://www.ietf.org/rfc/rfc2119.txt>
- [WAP] "Wireless Application Protocol Architecture Specification, WAP Forum, 30-Apr-1998. URL: <http://www.wapforum.org/>
- [WMLScript] "WMLScript Language Specification", WAP Forum, 17-Jun-1999. URL: <http://www.wapforum.org/>
- [WTA] WAP-169, "Wireless Telephony Application Specification", WAP Forum. URL: <http://www.wapforum.org/>
- [WTAI] WAP-170, "Wireless Telephony Application Interface Specification", WAP Forum. 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
ANSI 136	TDMA Cellular/PCS – Radio Interface – Mobile Station – Base Station Compatibility Standard
RFC	Request For Comments
URI	Uniform Resource Identifier [RFC2396]
WAP	Wireless Application Protocol [WAP]
WTA	Wireless Telephony Applications [WTA]
WTAI	Wireless Telephony Applications Interface [WTAI]

5 ANSI 136 Specific Library

In addition to the WTAI functions defined in [WTAI], ANSI 136 networks also support the functions specified in this chapter.

5.1 WTA Events

These events are related to ANSI 136 devices. All WTA event parameters are conveyed as strings.

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].

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 WMLScript Function Libraries

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 IDs*.

Table 1 , WMLScript Functions

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

Appendix B Static Conformance Requirements

This static conformance clause defines a minimum set of features that should be implemented to ensure that WTA could interact with the mobile network. A feature can be optional or mandatory.

B.1 Client features

B.1.1 WTA Events

Item	WTA Event	Reference	Status
WTAI_ANSI136E_C001	IncomingAlert (wtaev-ansi136/ia)	5.1.1	M
WTAI_ANSI136E_C002	IncomingFlash (wtaev-ansi136/if)	5.1.2	M

B.1.2 WMLScript Functions

Item	Function	Reference	Status
WTAI_ANSI136S_C001	WTAANSI136.sendFlash	5.2.1	M
WTAI_ANSI136S_C002	WTAANSI136.sendAlert	5.2.2	M

B.1.3 WMLScript Bytecode Interpreter Capabilities

Item	Function	Reference	Status
WTAI_ANSI136INT_C001	Supports ANSI 136 Network WTAI library identifier	A	M
WTAI_ANSI136INT_C002	Supports ANSI 136 Network WTAI function identifiers	A	M

B.2 Server features

B.2.1 WMLScript Encoder Capabilities

Item	Function	Reference	Status
WTAI_ANSI136ENC_S001	Supports ANSI 136 Network WTAI library identifier	A	M
WTAI_ANSI136ENC_S002	Supports ANSI 136 Network WTAI function identifiers	A	M

Appendix C Specification-track Document History

Document: Wireless Telephony Application Interface, ANSI 136 Specific Addendum (WTAI ANSI 136)

Document Identifier: WAP-172

Base Specification Approval Date: November, 1999

SINs Incorporated in this baseline document:

SIN Approval Date	SIN Document Identifier
June, 2000	WAP-172_100