

WAP Specification Information Note

WAP-170_102-WTAI

Version 26-Feb-2001

**A Wireless Application Protocol
Specification Information Note for:**

**Wireless Telephony Application Interface Specification
WAP-170-WTAI
Version 07-Jul-2000**

Notice:

© Wireless Application Protocol Forum, Ltd. 2001.

Terms and conditions of use are available from the Wireless
Application Protocol Forum Ltd. Web site
(<http://www.wapforum.org/what/copyright.htm>).

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. CHANGE 1 USE OF TONE_SEQUENCE IN WTAI SENDDTMF CALLS	5
3.1 CHANGE CLASSIFICATION	5
3.2 CHANGE SUMMARY	5
3.3 CHANGE	5
4. CHANGE 2 FIX A NUMBER OF ERRORS IN THE JUNE 2000 WTAI SPECIFICATION.....	7
4.1 CHANGE CLASSIFICATION	7
4.2 CHANGE SUMMARY	8
4.3 CHANGE	8

1. Scope

This Specification Information Note (SIN) covers changes against the 07-Jul-2000 version of the WTAI specification.

- Replacement of the dialstring with the tone_sequence in the WTAI functions WTAIPublic.sendDTMF, WTAIVoiceCall.sendDTMF, the URI call wtai://wp/sd and in the WTA event DTMFSent. Tone_sequence is defined in the WAP General Formats document.
- Fix a number of errors in the June 2000 WTAI specification. Disambiguated return value for functions which take phone number as a direct parameter and the phone number passed does not conform to the WAP General Formats Spec.

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 Forum Ltd, 2001.

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. Change 1 Use of tone_sequence in WTAI sendDTMF calls

3.1 Change Classification

1 – New Feature, Major Change or Market Effecting Change	[]
2 – Bug Fixes	[X]
3 – Clerical Corrections	[]

3.2 Change Summary

The WTAI library function “sendDTMF” currently accepts a parameter defined by dialstring in the General Formats specification. However, dialstring allows a string with a plus sign (“+”) in it and there is no defined mapping or generally accepted mapping of the plus sign into DTMF. The desired input to the sendDTMF function is a sequence of DTMF tones and pauses (absence of tones). This desired input is expressed in the General Formats variable tone_sequence which was introduced by the approved WAP-GF_CR_HP_2000-OCT-09_1 change request. This change should only be incorporated in **WAP-170-WTAI-20000707-A** specification if the SIN introducing WAP-GF_CR_HP_2000-OCT-09_1 change to WAP-188, “WAP General Formats Document” is approved.

This change replaces the dialstring with the tone_sequence in the WTAI functions WTAIPublic.sendDTMF, WTAIVoiceCall.sendDTMF, the URI call wtai://wp/sd and in the WTA event DTMFSent.

3.3 Change

8.2.2 WTAIPublic.sendDTMF

Function:	sendDTMF(<i>dtmf</i>)
Function ID:	1
Description:	<p>Sends a DTMF sequence through the voice call most recently created using the WTAIPublic.makeCall or wtai://wp/mc function. This function is blocking. No WTA events are generated as a direct or indirect result of invoking this function.</p> <p>The <i>dtmf</i> parameter specifies the DTMF sequence to be sent and must be a dialstring tone_sequence as defined in [FORMAT].</p> <p>This function returns an empty string if the DTMF sequence was sent successfully, returns an error-code that indicates an error in sending the DTMF sequence, or returns invalid if the function fails.</p>
Permission Types:	CONTEXT, SINGLE (see [WTA]).
Parameters:	<i>dtmf</i> = string (dialstring tone_sequence)
Return value:	empty string or invalid or error-code (must be one of the following decimal values: -108 = no active voice connection

-1 = unspecified error)

Associated Events: -

Exceptions: If the *dtmf* parameter is not a [tone_sequence_dialstring](#) as defined in [FORMAT], this function returns *invalid*.

Example: `var flag = WTAPublic.sendDTMF("555*1234");`

8.3.2 wtai://wp/sd

Function: `wtai://wp/sd;dtmf`

Description: Sends DTMF sequence through the voice call most recently created using the `WTAPublic.makeCall` or `wtai://wp/mc` function. This function is blocking. No WTA events are generated as a direct or indirect result of invoking this function.

The *dtmf* parameter specifies the DTMF sequence to be sent and must be a [tone_sequence_dialstring](#) as defined in [FORMAT].

This function returns a value that indicates the result of the function invocation; i.e., whether or not the DTMF was sent.

Permission Types: CONTEXT, SINGLE (see [WTA]).

Parameters: `dtmf = string (tone_sequence_dialstring)`

Result value:

- "" = DTMF sequence was sent
- "-108" = could not send DTMF because no active voice connection
- "-200" = invocation error
- "-1" = unspecified error

Associated Events: -

Exceptions: If the *dtmf* parameter is not a [tone_sequence_dialstring](#) as defined in [FORMAT], this function returns an invocation error.

If the DTMF can not be sent, e.g., because there is no active call, this function returns an invocation error.

Example: `<go href="wtai://wp/sd;555*1234"/>`

9.1.6 wtaev-cc/dtmf

Event Name: `DTMFSent`

Event ID: `wtaev-cc/dtmf`

Parameters: `callHandle, dtmf`

Description: Indicates the DTMF sequence has been sent on a voice call. A sequence of invocations of `WTAVoiceCall.sendDTMF()` will generate a matching sequence of `DTMFSent` events in order, except in cases when the DTMF can not be sent, such as if the call is cleared.

The *callHandle* parameter contains the call handle for the call as defined in section 5.2.2.

The *dtmf* parameter contains the DTMF sequence that was sent and must be a [tone_sequence_dialstring](#) as defined in [FORMAT].

9.2.4 WTAVoiceCall.sendDTMF

Function: `sendDTMF(callHandle,dtmf)`

Function ID: 3

Description: Sends a DTMF sequence through a voice call. This function is non-blocking. Subsequent WTA events signal that the DTMF has been sent on the voice call.

The *callHandle* parameter identifies which call is to receive the DTMF. (See section 5.2.2 for a description of the call handle.)

The *dtmf* parameter specifies the DTMF sequence to be sent and must be a [tone_sequence_dialstring](#) as defined in [FORMAT].

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

dtmf = `string` ([tone_sequence_dialstring](#))

Return value: empty `string` or `invalid`

Associated Events: Exactly one `DTMFSent` event occurs when the DTMF sequence has been sent on the voice call.

Exceptions: If the *callHandle* parameter does not refer to a voice call through which DTMF may be sent, this function returns `invalid`.

If the *dtmf* parameter is not a [tone_sequence_dialstring](#) as defined in [FORMAT], this function returns `invalid`.

Example: `var flag = WTAVoiceCall.sendDTMF(handle,"555*1234");`

4. Change 2 Fix a number of errors in the June 2000 WTAI specification

4.1 Change Classification

1 – New Feature, Major Change or Market Effecting Change	[]
2 – Bug Fixes	[X]
3 – Clerical Corrections	[]

4.2 Change Summary

- Disambiguated return value for functions which take phone number as a **direct** parameter and the phone number passed does not conform to the WAP General Formats Spec. This change is not applied to WTAPhoneBook.change() as the phone number is passed indirectly.
- Changed wording SESSION to CONTEXT for Permission Types.

4.3 Change

8.2.3 WTAPublic.addPBEntry

Function: addPBEntry(*number*,*name*)

Function ID: 2

Description: Writes a new phonebook entry.

The *number* parameter specifies the phone number to associate with the entry and must be a phone-number as defined in [FORMAT].

The *name* parameter specifies the name to associate with the entry and may be an empty string.

This function returns an empty string if the phonebook entry was added successfully, returns an error-code that indicates an error in adding the phonebook entry, or returns invalid if the function fails.

Permission Types: SINGLE (see [WTA]). The *name* and *number* parameters must be displayed to the user when obtaining permission.

Parameters: *number* = string (phone-number)

name = string

Return value: empty string or invalid or error-code (must be one of the following decimal values:

-100 = *name* parameter is unacceptable or too long

~~-101 = *number* parameter is not a valid phone number~~

-102 = *number* parameter is too long

-103 = phonebook entry could not be written

-104 = phonebook is full

-1 = unspecified error)

Associated Events: -

Exceptions: [- If the *number* parameter is not a phone-number as defined in \[FORMAT\], this function returns invalid.](#)

Example: var flag = WTAPublic.addPBEntry("5554367", "EINSTEIN");

8.3 URI functions

8.3.1 wtai://wp/mc

Function: wtai://wp/mc;*number*![result](#)

Description:	<p>Initiates a mobile-originated voice call. This function is blocking. No WTA events are generated by the voice call initiated using this function.</p> <p>The <i>number</i> parameter specifies the destination to call and must be a phone-number as defined in [FORMAT].</p> <p>This function returns a value that indicates the result of the function invocation, i.e., whether or not the voice call was established.</p>
Permission Types:	SINGLE (see [WTA]). The <i>number</i> parameter must be displayed to the user when obtaining permission.
Parameters:	<i>number</i> = string (phone-number)
Result value:	<p>"" = call was established successfully</p> <p>"-105" = called party is busy</p> <p>"-106" = network is not available</p> <p>"-107" = called party did not answer</p> <p>"-200" = invocation error</p> <p>"-1" = unspecified error</p>
Associated Events:	-
Exceptions:	<p>If the <i>number</i> parameter is not a phone-number as defined in [FORMAT], this function returns an invocation error.</p> <p>If the voice call can not be initiated, e.g., because there is no service, this function returns an invocation error.</p>
Example:	<go href="wtai://wp/mc;5554367! resultvar " />

8.3.2 wtai://wp/sd

Function:	wtai://wp/sd;dtmf! result
Description:	<p>Sends DTMF sequence through the voice call most recently created using the WTAPublic.makeCall or wtai://wp/mc function. This function is blocking. No WTA events are generated as a direct or indirect result of invoking this function.</p> <p>The <i>dtmf</i> parameter specifies the DTMF sequence to be sent and must be a dialstring as defined in [FORMAT].</p> <p>This function returns a value that indicates the result of the function invocation; i.e., whether or not the DTMF was sent.</p>
Permission Types:	CONTEXT, SINGLE (see [WTA]).
Parameters:	<i>dtmf</i> = string (dialstring)
Result value:	<p>"" = DTMF sequence was sent</p> <p>"-108" = could not send DTMF because no active voice connection</p> <p>"-200" = invocation error</p> <p>"-1" = unspecified error</p>
Associated Events:	-
Exceptions:	If the <i>dtmf</i> parameter is not a dialstring as defined in [FORMAT], this function returns an invocation error.

~~If the DTMF can not be sent, e.g., because there is no active call, this function returns an invocation error.~~

Example: `<go href="wtai://wp/sd;555*1234!resultvar" />`

8.3.3 wtai://wp/ap

Function: `wtai://wp/ap;number;name!result`

Description: Writes a new phonebook entry.

The *number* parameter specifies the phone number to associate with the entry and must be a phone-number as defined in [FORMAT].

The *name* parameter specifies the name to associate with the entry and may be an empty string.

This function returns a value that indicates the result of the function invocation; i.e., whether or not the phonebook entry was added.

Permission Types: SINGLE (see [WTA]). The *name* and *number* parameters must be displayed to the user when obtaining permission.

Parameters: *number* = string (phone-number)

name = string

Result value: "" = phonebook entry was added

"-100" = *name* parameter is unacceptable or too long

~~"-101" = *number* parameter is not a valid phone number~~

"-102" = *number* parameter is too long

"-103" = phonebook entry could not be written

"-104" = phonebook is full

"-200" = invocation error

"-1" = unspecified error

Associated Events: -

Exceptions: If the *number* parameter is not a phone-number as defined in [FORMAT], this function returns an invocation error.

~~If the phonebook entry is not written, e.g., because the phonebook is full, this function returns an invocation error.~~

Example: `<go href="wtai://wp/ap;5554367;EINSTEIN!resultvar" />`

11.2.1 WTAPhoneBook.write

Function: `write(index,number,name)`

Function ID: 0

Description: Writes a phonebook entry, overwriting any existing entry.

The *index* parameter specifies the location of the entry within the phonebook. (See section 5.4.1 for a description of the phonebook index.) If the *index* parameter is zero, then the entry is stored in a location that is currently empty; the choice of location is implementation dependent.

The *number* parameter specifies the phone number to associate with the entry and must be a phone-number as defined in [FORMAT]. It becomes the "number" field of the phonebook entry.

The *name* parameter specifies the name to associate with the entry and may be an empty string. It becomes the "name" field of the phonebook entry.

This function returns the index where the entry was stored if successful, or returns an error-code under certain conditions, or returns *invalid* if the function fails.

Permission Types: BLANKET, [CONTEXT](#)SESSION, SINGLE (see [WTA]).

Parameters:

- index* = integer
- number* = string (phone-number)
- name* = string (associated name)

Return value: integer or *invalid* or error-code.
error-code must be one of the following decimal values:

- 100 = *name* parameter contains unacceptable characters or is too long
- ~~-101 = *number* parameter does not contain a phone number~~
- 102 = *number* parameter is too long
- 103 = entry could not be written
- 104 = phone book is full
- 1 = unspecified error

Associated Events: -

Exceptions: If the *index* parameter is unacceptable, e.g., does not refer to a valid phonebook entry location, this function returns *invalid*.

[If the *number* parameter is not a phone-number as defined in \[FORMAT\], this function returns *invalid*.](#)

Example: var index = WTAPhoneBook.write(0,"5554367", "A. Einstein");

11.2.2 WTAPhoneBook.search

Function: search(*field*,*value*)

Function ID: 1

Description: Returns the index of a phonebook entry that matches the given search criteria. The index of an empty phonebook entry is never returned.

The *field* parameter specifies which field of the phone book entry to inspect.

The *value* parameter specifies the desired value of that field, where the empty string is a wildcard.

To start a search, this function must be invoked with a non-empty string for the *field* parameter. If the *value* parameter is a non-empty string, the search is initialized to return the index of all phone book entries containing that string anywhere within the field value, using case-insensitive comparison. If the *value* parameter is an empty string, the search is initialized to return the index of all non-empty phone book entries that have a non-empty string for the value of the specified field.

NOTE: All implementations must perform at least a literal string comparison between the *value* parameter and the phonebook entry field value. However, an implementation may use an enhanced string comparison algorithm, such as one that matches characters with diacritical marks to unadorned characters (e.g., "e" matches "e é ê ë e").

To continue a search, this function must be invoked with an empty string for the *field* and *value* parameters.

Each invocation of this function, including the one that initializes the search, returns an index of a phonebook entry that meets the search criteria.

This function returns the next index of a phonebook entry that matches the search criteria, or returns *invalid* if there is not another search result (e.g., the end of the list was reached) or if the function fails.

Note: This function is intended for use within a loop with a non-empty string for the *field* parameter on the first invocation, with the empty string for the input parameters of each subsequent time through the loop, and with the loop exiting when *invalid* is returned.

Permission Types: BLANKET, [CONTEXT](#)SESSION, SINGLE (see [WTA]).

Parameters: *field* = string (field name or empty)
value = string (field value or empty)

Return value: integer or *invalid*

Associated Events: -

Exceptions: If the *field* parameter is the empty string and the *value* parameter is not (i.e., the function is invoked illegally), this function returns *invalid*.

If the *field* parameter is unrecognized, unsupported, or otherwise unavailable, this function returns *invalid*.

If this function is invoked to continue a search without the search criteria first having been initialized, then this function returns *invalid*.

Example: var index = WTAPhoneBook.search("name", "alb");

11.2.3 WTAPhoneBook.remove

Function: remove(*index*)

Function ID: 2

Description: Removes a phonebook entry, that is, makes the phonebook entry an empty entry.

The *index* parameter specifies the location of the entry within the phonebook. (See section 5.4.1 for a description of the phonebook index.)

This function returns an empty string if the function is successful, returns an error-code under certain conditions, or returns invalid if the function fails.

Permission Types: BLANKET, [CONTEXT](#)SESSION, SINGLE (see [WTA]).

Parameters: *index* = integer (index of location to remove)

Return value: empty string or invalid or error-code.

error-code must be one of the following decimal values:

-105 = could not remove phonebook entry

-1 = unspecified error

Associated Events: -

Exceptions: If the *index* parameter is unacceptable, e.g., does not refer to a valid, non-empty phonebook entry, this function returns invalid.

Example: var flag = WTAPhoneBook.remove(index);

11.2.4 WTAPhoneBook.getFieldValue

Function: getFieldValue(*index*,*field*)

Function ID: 3

Description: Retrieves a field value from a specific phonebook entry.

The *index* parameter specifies the location of the entry within the phonebook. (See section 5.4.1 for a description of the phonebook index.)

The *field* parameter specifies the name of the field to retrieve from the phonebook entry info. (See section 5.4.2 for a description of the phonebook entry fields.)

This function returns a value that depends on the *field* parameter. Any string may be given for the *field* parameter, however, if the *field* parameter matches one of the field names specified in section 5.4.2, the corresponding information must be returned as specified in that section. If the *field* parameter is unrecognized, unsupported, or otherwise unavailable, an empty string is returned.

Permission Types: BLANKET, [CONTEXT](#)SESSION, SINGLE (see [WTA]).

Parameters: *index* = integer (index of entry of interest)

field = string (name of field to retrieve)

Return value: any except invalid (field value), or invalid (failure indication)

Associated Events: -

Exceptions: If the *index* parameter is unacceptable, e.g., does not refer to a valid, non-empty phonebook entry, this function returns invalid.

Example: var number = WTAPhoneBook.getFieldValue(index, "number");

11.2.5 WTAPhoneBook.change

Function: `change(index,field,newValue)`

Function ID: 4

Description: Store the given value in the specified field in a phonebook entry, overwriting the existing value if any.

The *index* parameter specifies the location of the entry within the phonebook. (See section 5.4.1 for a description of the phonebook index.)

The *field* parameter specifies the name of the field to change within the phonebook entry. (See section 5.4.2 for a description of the phonebook entry fields.)

The *newValue* parameter specifies the new value for the specified field within the phonebook entry. (See section 5.4.2 for a description of the phonebook entry fields.)

This function returns an empty string if the function is successful, returns an error-code under certain conditions, or returns `invalid` if the function fails.

Permission Types: BLANKET, [CONTEXTSESSION](#), SINGLE (see [WTA]).

Parameters:

- index* = integer (location of entry)
- field* = string (name of field to change)
- newValue* = string (new value for the field)

Return value: empty string or `invalid` or error-code
error-code must be one of the following decimal values:

- 100 = *field* parameter is "name" and *newValue* contains an unacceptable character or is too long
- 101 = *field* parameter is "number" and *newValue* is not a phone-number
- 102 = *field* parameter is "number" and *newValue* is too long
- 103 = entry could not be changed
- 104 = phonebook is full
- 106 = *field* parameter is not supported
- 107 = *field* parameter is not "name" or "number" and *newValue* is unacceptable or is too long
- 1 = unspecified error

Associated Events: -

Exceptions: If the *index* parameter is unacceptable, e.g., does not refer to a valid, non-empty phonebook entry location, this function returns `invalid`.

Example: `var flag = WTAPhoneBook.change(index, "name", "ALBERT");`