

第9章 无线电话应用接口规范

9.1 范围

无线应用协议 (Wireless Application Protocol , WAP) 是WAP论坛经过不断努力得到的成果, 它提供了一个业界技术规范, 以便开发出适用于各种无线通信网络的应用和业务。 WAP论坛的工作范围就是为各种业务和应用制定一系列的技术规范。无线市场正在快速增长, 新的用户不断增多, 新的业务不断涌现。为了给运营商和生产商提供一个面对先进业务、多种类业务和快速灵活业务生成的商机, WAP制定了一系列用于传输层、会话层和应用层的协议。有关WAP体系结构更多的信息, 请参阅“ 无线应用协议体系结构规范 ” (Wireless Application Protocol Architecture Specification) [WAP]。

本规范是WAP无线应用环境 (WAE) 的一个扩展集, 用来支持无线电话应用。无线电话应用的特性以接口的形式引入, 规范中使用的首字母缩略词 WTAI表示无线电话应用接口。为了更好的理解这个规范, 读者应对 WML [WML]和WMLScript [WML Script]有一定程度了解。

9.2 WTA 研究背景

WAP的WTAI特性提供了创建电话应用的方法。借助有关的 WTAI函数库, WTA用户代理可以实现电话应用。一个典型的例子是利用 WTAI的函数, 从一个 WML页面(deck)、卡片(card)或者WALScript中建立由移动终端发起的呼叫。

WTA 应用模型建立在一个运行 WML和WML Script程序的 WTA用户代理上。WTA用户代理通过 WTAI函数库实现与网络服务有关的函数调用, 它可以接收从移动网络发来的 WTA事件以及从WTA服务器推过来的内容 (如 WML页面和WTA事件等)。WTA事件和WTAI函数为对移动网络中的资源 (如呼叫控制) 进行操作和处理提供了可能性。

通过对WML和WMLScript内容的推操作, WTA服务器能够动态地激活应用。

9.2.1 WTAI函数库

通过对WTAI性能分类, 得到了一个 WTAI函数库的集合。一个函数的归属库与它的类型及使用率有关。对 WTAI函数库的访问, 可以从 WML中利用URL完成, 还可以从 WMLScript中利用脚本函数库完成。

通用网络的WTA 包含了所有网络中最常用的特性。只有 WTA的用户代理才能访问这些函数。例如实现呼叫建立和来话应答的函数。

特殊网络的WTA 包含了只能在特殊网络中使用的特性, 特殊的操作特性也被归在这个集合中。

公共的WTA 包括一些简单的特性。通过标准的 WAE用户代理, 第三方正在运行的应用程序可以访问这些特性。

9.2.2 事件处理

WTA事件是一种方法，用这种方法可以传递 WTA服务器或移动网络中的状态变化。对于一个WTA事件，可以设置用户代理要进行的操作，WTA事件必须与 URL指定的处理这个WTA事件的内容相匹配。对 WTA事件的处理，既可以使用 WTA事件表，也可以在用户代理的上下文中利用临时的事件绑定动态地完成。

有关事件处理的详细信息，请参阅 WTA规范[WTA]。

9.3 WTA接口

9.3.1 WTAI函数库

根据函数的类型，WTAI函数被划分到各个函数库中。函数库可以用在一个特殊类型的专门网络中，这时，在这个函数库的名称中，就包含有这个“知名”网络的名字。针对公共和通用网络的WTA，WTAI规范预先定义了一个 WTA函数库（见表 9-1和表9-2）。在9.10节中，给出了特殊网络WTA函数库的定义。

表9-1 公共的WTA函数库

函 数 库	名 称	函数库描述
公共的 WTAI	“ wp ”	可用的公共 WTAI函数

表9-2 通用网络的WTA函数库

函 数 库	名 称	函数库描述
呼叫控制	“ cc ”	呼叫控制库，在一个呼叫过程中，完成呼叫建立和设备的控制
网络文本	“ nt ”	网络文本库，发送和取回网络文本
电话簿	“ pb ”	电话簿库，管理设备电话簿的接入
其他的库	“ ms ”	处理其他的特性，例如逻辑指示

9.3.2 WTAI 的API 间隔符

除非特别说明，假设所有的参数都是字符串。此外，如果没有标明可以选择，所有的参数都是必须的。访问 WTAI函数，可以利用 WTAI的 URI方案（这是一种类似于 CGI 风格的参数调用方案），也可以利用 WMLScript定义的调用方案。

1. WTAI语法中使用的记号

- < > 尖括号表示一个枚举参数。
- [] 方括号表示可选项。
- | 垂直线表示相互排斥的选项。
- () * 不重复或者重复多次。
- * () 重复一次或者重复多次。

2. 参数规定

除非特别声明，一般的情况下给出所有的输入和输出参数。如果没有规定结果参数，

WTA的用户代理也不会出错。在这种情况下，推荐的处理过程是丢弃运算结果。

9.3.3 WTAI的URI方案

在这种方案中，是使用专门的 WTAI URI 编码方案（见表 9-3），通过 URI “调用”，从 WML中访问WTAI函数库，URI由预先定义的 WTAI函数库的引用和实际的函数名一起构成。一个函数库可以用WTAI 的URI库标识符标识。例如，一个预先定义的库是“ WTACallCont ”，它定义了常用的呼叫控制特性。

对于特殊网络的特性，WTAI的URI用一套“知名”的网络库名定义。
WTAI函数使用 URI 命名，[RFC1630]给出了URI的定义和使用的字符集。这样，对于 WTAI的URI中使用的一些特殊字符（如空格符）必须进行转义（有关转义的详细信息，请参阅[RFC1630]）。

结构 wtai://<library>/<function> (; <parameter>) * [! <result> (; <result>) *]

表9-3 WTAI 的URI方案

<library> 函数库	库名，用于标识函数的类型。例如，呼叫控制使用的库名是“ cc ”
<function> 函数	给定库中的函数标识符。例如，在“ 通用网络的 WTA ”中，“ ac ”库的函数“ Accept Call ”
<parameter>参数	传递给函数的零个或多个参数。使用分号“ ; ”作为相继参数的分隔符
<result> 结果	结果数据的开始部分用一个惊叹号“ ! ”表示。零个或多个变量名存储在 WTA 用户代理的上下文中，作为函数调用的结果。使用分号“ ; ”作为相继结果数据的分隔符

9.3.4 WTAI函数定义格式

描述	在这里描述函数。
URI	函数的URI格式。
WMLScript	函数的WMLScript格式。
函数标识符（ ID ）	函数在库中的编号。
参数	描述给出的参数。
输出	描述函数输出。关于 WTAI已定义的错误代码，见9.11节。
示例	给出通过URI方案调用函数的示例。 给出通过脚本方案调用函数的示例。
相关的事件	这一部分列出了函数调用之后，可发生的 WTA的事件列表。
注释	额外的帮助注释。

9.4 公共的WTA

公共的WTA函数可用在非WTA网络服务器产生的应用（也就是第三方服务器产生的应用）中。在网络的WTA中，在WTA运行之前用户能够删除任何给定的操作，而公共的应用对这些函数的处理与网络的WTA不同。例如，客户服务中的“ 电话号码指南 ”，是一个由第三方提供的应用，其上列出的电话号码实际上是“ 标识符 ” URI，用这个标识符，可以调用“ 公共的

WTA ” 函数 “ makeCall ”。	
名称	WTAPublic
库标识符（ ID ）	512
描述	这个库包含一个公共函数，该函数可以显示拨打的号码。
• 建立通话	
描述	通过拨打特定的号码，这个函数可以建立由移动终端发起的呼叫，用户应该清楚地了解这个操作。 MakeCall函数不仅能用在 WTA中，还可以用在任何应用程序中，用来向用户显示拨打的电话号码
URI	wtai://wp/mc;< number >
WMLScript	makeCall(<i>number</i>);
函数ID	0
参数	<number> = 字符串： 呼叫的目的号码，可以是任何有效的电话号码字符和数字。
输出	—
示例	URI:wtai//wp/mc;5554367 WMLScript:WTAPublic.makeCall(“ 5554367 ”);
相关事件	—
注释	这个呼叫必须使用标准的 MMI结束。

9.5 通用网络的WTA

本节定义的函数可以在 WAP计划使用的各种移动网络中使用。在9.10节中，给出了特殊网络的函数定义。

9.5.1 网络事件

WTAI规定了与移动网络相匹配的 WTA事件的名，即本地事件（见表 9-4）。这些移动网络事件传递移动网中的服务状态，并可以由活动的上下文处理，或者用来开始一个有新的上下文的WTA用户代理。

表9-4 本地事件

事 件	参 数	描 述
cc/ic	id, callerID	来电指示。一个来话呼叫已经到达用户代理，可以用 WTA的 “ Accept Call ” 函数，从应用中截获这个呼叫 <id>:用户代理自己生成的标识，用于后面的呼叫控制操作 <CallerID>:如果用户代理调用，包含了始呼方的电话号码；否则，返回一个空字符串
cc/di	id	断开连接指示。网络端已经断开了这个通话的连接 <id>:连接被断开的通话标识
cc/ca	id, callerID	呼叫应答，被叫方已经拿起了话筒 <id>:呼叫的标识，这个呼叫已被应答 <CallerID>:如果用户代理调用，包含被叫方的电话号码；否则，返回一个空字符串

(续)

事 件	参 数	描 述
cc/cn	id, result	呼叫没有应答，也就是被叫方没有应答 <id>:占线的呼叫标识 <result>:这个结果指出被叫方没有应答的原因 请参阅9.11节“预先确定的 WTAI 错误代码”

9.5.2 呼叫控制

在一个呼叫过程中，下面的 WTA 函数可以用来控制诸如接收呼叫和释放呼叫这样的特性。

名称

WTACallControl

函数库标识ID

513

描述

这个函数库包含了与呼叫控制有关的函数。对于所有的“知名”网络，这些函数是常见的。

1. 建立通话

描述

用给定的电话号码，建立一个由移动终端发起的呼叫。参数 *mode* 指出：如果 WTA 用户代理的上下文结束，应该如何处理这个呼叫。有两种模式：“放弃”和“保持”。“放弃”意味着如果上下文需要重新开始，OS 释放这个呼叫；“保持”意味着即使当前的上下文已经结束，也要保持这个呼叫。

URI

wtai://cc/sc;< number>;< mode > [! <result>]

WMLScript

Set-up(number,mode);

函数ID

0

参数

< Number> = 字符串：
呼叫的目的号码，可以使用任何有效的电话号码字符和数字。

< mode> :
0 = 放弃，在当前的上下文被删除时，放弃呼叫。
1 = 保持，在当前的上下文被删除时，保持呼叫。

输出

< result> = 字符串：
返回值是被创建呼叫的标识，在呼叫失败的情况下返回一个负数，作为 WTAI 的错误代码。

示例

URI:wtai://cc/sc;5554367;1
WMLScript:WTACallCont.setup(“ 5554367 ”,0);

相关事件

cc/di，呼叫连接断开。
cc/ca，呼叫应答。
cc/cn，呼叫没有应答。

注释

—

2. 接受呼叫

描述

接受一个来话呼叫，或者是等待通话和拿起话筒。在这里，“id”是由内部的呼叫处理设备分配的一个顺序号，如果进行通话，就返回这个

顺序号；如果由于某种原因，这个通话不能进行，返回值包含一个错误代码。

任何一方或者网络都可以结束这个通话。当通话结束时，产生一个连接断开指示，应用程序可以检测到这个指示。

如果WTA用户代理的上下文结束，参数 *mode* 指出应该如何处理这个通话。有两种模式：“放弃”和“保持”。“放弃”意味着如果上下文需要重新开始，OS释放这个呼叫；“保持”意味着即使当前的上下文已经结束，也要保持这个呼叫。

URI	wtai://cc/ac ; <id> ; <mode>[! <result>]
WMLScript	accept(id,mode);
函数ID	1
参数	< id> = 字符串： 被接受的呼叫的标识。 < mode> : 0 = 放弃，在当前的上下文被删除时，放弃呼叫。 1 = 保持，在当前的上下文被删除时，保持呼叫。
输出	< result> = 字符串： 返回值或是被创建呼叫的标识，在呼叫失败的情况下，返回一个负数，作为WTAI的错误代码。
示例	URI:wtai//cc/ac;1;1 WMLScript:WTACallCont.accept(“ 1 ”,0);
相关事件	cc/di，呼叫连接断开。 cc/ca，呼叫应答。 cc/cn，呼叫没有应答。
注释	—
3. 呼叫释放	
描述	释放给定的呼叫。涉及到多方通话的呼叫，可以用呼叫标识释放。
URI	wtai://cc/rc;< id>[! <result>]
WMLScript	release(id);
函数ID	2
参数	< id> = 字符串： 被释放呼叫的标识。
输出	< result> = 字符串： 返回值是被创建呼叫的标识，在呼叫失败的情况下，返回一个负数，作为WTAI的错误代码。
示例	URI:wtai//cc/rc;1 WMLScript:WTACallCont.release(“ 1 ”);
相关事件	cc/di，呼叫连接断开
注释	—

4. 发送DTMF拨号音

描述	通过一个激活的语音连接，发送DTMF拨号音序列。如果呼叫成功，返回的整数值是0；在没有成功结果的情况下，返回错误代码。
URI	wtai://cc/sd;<dtmf>[! <result>]
WMLScript	sendDTMF(dtmf);
函数ID	3
参数	< dtmf > = 字符串： 任何有效的标准DTMF字符序列。
输出	< result> = 字符串： 小于0的整数值表示没有成功的运行。
示例	URI:wtai://cc/sd;555*1234 WMLScript:WTACallCont.sendDTMF(" 555*1234 ");
相关事件	cc/di，呼叫连接断开。
注释	—

9.5.3 网络文本

网络文本 WTA 函数库处理从设备的网络文本应用中发送和取回的文本信息，有用的“网络文本”函数是 send、read、delete 和 getFieldValue 函数。

名称	WTANetText
库标识符 (ID)	514
描述	这个库包含了处理网络文本发送和取回的函数。
1. 发送文本	
描述	如果在网络中可以使用这个特性，向 number 标识的目的地发送网络文本信息。
URI	—
WMLScript	Send (number,text);
函数ID	0
参数	< number> = 字符串： 目的地号码，可以是任何有效的电话字符和数字。 < text > = 字符串： 网络文本数据结构，被发送的文本。
输出	< result> = 字符串： 小于0的整数值表示没有成功的运行。
示例	WMLScript:WTANetText.send(" 5554567 " , " WAP Forum ");
相关事件	—
注释：	—
2. 读文本	
描述	读存储在设备中的网络文本数据。数据以字段编码字符串的格式取回。使用 GetFieldValue 函数可以抽取任意给定的字段值。

URI	—
WMLScript	read(id);
函数ID	2
参数	< id> = 字符串： 要读取的网络文本标识。如果 id是空字符串，最后面的未读信息被返回。
输出	< struct> = 字符串： 用于接收网络文本数据结构变量的名称。小于 0的整数值表示没有成功运行。
示例	WMLScript:WTANetText.read(3);
相关事件	—
注释	—
3. 删除文本	
描述	删除由 id指定的网络文本信息。如果没有被确认的文本记录，返回一个错误代码。
URI	—
WMLScript	delete(id);
函数ID	3
参数	< id> = 字符串： 被删除的网络文本信息标识。
输出	< result> = 字符串： 小于0的整数值表示没有成功的运行。
示例	WMLScript:WTANetText.delete(3);
相关事件	—
注释	—
4. 获得字段值(GetFieldValue)	
描述	从字符串<struct>中取回由“ field ”标识的值。
URI	-wtai://nt/gv;<struct>;<field> !<result>
WMLScript	getFieldValue(struct, field);
函数ID	4
参数	< struct> = 字符串： 格式化的字符串包含了与数据相关的字段。 < field> = 字符串： 字段名包含了要从< struct>中取回的值。
输出	< result> = 字符串： 字符串是与请求字段有关的值。小于 0的整数值表示没有成功的运行。
示例	WMLScript:WTANetText.getFieldValue(\$struct, “ name ”);
相关事件	—

注释 如果在<struct>中这个字段不存在，结果包含一个空字符串。

9.5.4 电话簿

电话簿 WTA 函数库处理向电话簿应用发出的操作请求。请求的操作可以是存储或者检索电话簿的记录，也可以是用给定的号码、名称和标识搜索电话簿。

利用读函数，电话簿记录可以作为“struct”取回，“struct”是一种格式化的字符串，它包含了与数据相关的字段。这样，每个字段的数据就可以用 GetFieldValue 函数取回。为了获得最小的实现，可以利用三种“已知”的字段名：

- name：包含记录的名称。
- number：包含记录的电话号码。
- id：包含记录的id。

电话簿库函数是 write、read、delete 和 GetFieldValue 函数。

名称	WTAPhoneBook
库标识符 (ID)	515
描述	这个库包含了处理面向电话簿应用 [⊖] 的操作，如电话簿记录的存储和检索等。

1. 写电话簿记录

描述	向电话簿写一个新的记录，先前任何有相同标识的记录都会被覆盖。如果没有指定标识，使用下一个可用的电话簿记录，同时返回新的标识。 在无法成功运行的情况下，输出包含一个负数，作为 WTAI 的错误代码。
----	---

URI	—
WMLScript	write(id, number, name);
函数ID	0
参数	<id> = 字符串： 电话记录标识。 <number> = 字符串： 被存储的电话号码。 <name> = 字符串： 与电话号码有关的名称。
输出	< result> = 字符串： 电话簿记录标识。小于0的整数值表示出错。
示例	WMLScript:WTAPhoneBook.write(“ 2 ” , “ 5554367 ” , “ EINSTEIN ”);
相关事件	—
注释	—

2. 读电话簿记录

⊖ 电话簿应用与具体的实现有关，不属于WAP定义的范畴。

描述 返回指定的电话簿记录（可以采用标识匹配或电话号码匹配方式）。选择标准可以基于标识（根据标识可以找到电话簿数据库的记录）、基于电话号码或者基于任何可查询的文本（如名称）。如果参数是空串（“ ”），在下次匹配时，会再次使用最后的标准；如果发现搜索标准，返回一个空字符串。

URI —

WMLScript read(id, criteria);
(criteria可以是id号、文本或者是号码)

函数ID 1

参数 无参数：

下次搜索时，还使用最后用到的搜索标准。

<mode> = 字符串：

搜索模式(I= id, t = text , n = number)

<id> = 字符串：

电话记录标识

<text> = 字符串：

搜索文本，如名称。

<number> = 字符串：

电话号码。

输出 < struct> = 字符串：

编码的信息结构。

示例 WMLScript:WTAPhoneBook.read(“ t ” , “ My MOM ”);

相关事件 —

注释 —

3. 删除电话簿记录

描述 删除电话簿记录。如果调用成功，结果变量包含一个 0；否则，返回一个负数，作为 WTAI 的错误代码。

URI —

WMLScript Delete(id);

函数ID 2

参数 <id> = 字符串：

电话记录标识。

输出 < result> = 字符串：

如果成功，取0值；小于0的整数值表示没有成功的运行。

示例 WMLScript:WTAPhoneBook.delete(“ 2 ”);

相关事件 —

注释 —

4. 获得字段值 (GetFieldValue)

描述 从字符串 <struct> 中返回由 “ field ” 指定的值。

URI	—
WMLScript	GetFieldValue(struct, field);
函数ID	3
参数	<p><struct> = 字符串: 格式化的字符串, 它包含了与数据有关的字段。</p> <p><field> = 字符串: 字段名, 它包含从<struct>中要取回的值。</p>
输出	<p>< result> = 字符串: 字符串, 包含与请求字段有关的值。</p>
示例	WMLScript:WTAPhoneBook.getFieldValue(\$struct, " name ");
相关事件	—
注释	如果在<struct>中这个字段不存在, 那么结果包含一个空字符串。

9.5.5 其他库

WTA用户代理还使用了各种有效的函数。

名称 WTAMisc

库标识符 (ID) 516

描述 这个库包含了用于控制设备逻辑特性 (如逻辑指示) 的函数。

1. 指示

描述 打开或关闭逻辑指示。MMI中的显示与具体的实现有关, 例如, 一个逻辑指示可以是视频的或音频的, 也可以是音频和视频的, 还可以设置一个指示的显示, (如显示E-mail的个数)。

URI —

WMLScript indication(type,operation,count);

函数ID 0

参数

<type> = 字符串:

- 0 = 到来的语音呼叫
- 1 = 到来的数据呼叫
- 2 = 到来的传真呼叫
- 3 = 呼叫等待
- 4 = 被接收的文本
- 5 = 语音信箱提示
- 6 = 传真提示
- 7 = E-mail提示
- 8-15 = 其他的提示

<operation> = 字符串:

- 1 = 设置

激活所选的指示(比如开始振铃、开始动画等等)

- 2 = 重新设置

使指示器回到设置函数被调用之前的状态，或者在不曾作过设置的情况下，回到指示器的默认状态。

<count> = 字符串:

新的文本、语音、邮件等的号码。

输出	—
示例	WMLScript:WTAMisc.indication(5,1,3);
相关事件	—
注释	WTA用户代理可以不显示计数，然而如何计数与具体的实现有关。

2. 终止WTA的用户代理

描述 这个函数删除内容，并结束用户代理的上下文。

URI wtai://ms/ec

WMLScript endcontext;

函数ID 1

参数 —

输出 —

示例 URI:wtai://ms/ec

WMLScript:WTAMisc.endcontext;

相关事件 —

注释 当只需要清除上下文时，可以使用[WML]中定义的newcontext属性。

9.6 术语定义

本规范中使用的术语和习惯用法如下：

客户端 (Client) 客户端是向服务器发出连接请求的设备或应用程序。

内容 (Content) 资源的同义词。

页面: (Deck) 一个WML文档，它可以包含WMLScript。

设备 (Device) 一个网络实体，能够发送和接收信息包的，并且有一个唯一的地址。在一个给定的上下文或跨越多重上下文，一个设备既可作为客户端，也可作为服务器。例如，一个设备作为其他服务器的客户端时可充当其他客户端的服务器。

服务器 (Server) 它是一种被动地等待一个或多个客户端连接请求的设备 (或应用程序)。服务器可以接受或拒绝来自客户端的连接请求。

用户 (User) 用户是一个通过用户代理观看、聆听或使用资源的人。

用户代理 (User Agent) 用户代理是可以解释WML、WMLScript 或其他内容的软件和设备。包括文本浏览器、语音浏览器和搜索引擎等。

无线标记语言 (WML) 无线标记语言是一种超文本标记语言，用于表示在窄带设备 (如：电话) 中传输的信息。

无线标记语言脚本 (WMLScript) 用来对移动设备进行编程的一种脚本语言。WMLScript 是JavaScript脚本语言的扩展子集。

9.7 缩略语

本规范采用了下列缩略语：

API	Application Programming Interface	应用编程接口
CGI	Common Gateway Interface	通用网关接口
DCS	Digital Communications System	数字通信系统
GSM	Global System for Mobile Communication	全球移动通信系统
OS	Operating System	操作系统
PCS	Personal Communications System	个人通信系统
PDC	Personal Digital Cellular System	个人数字蜂窝系统
RFC	Request For Comments	请求注释
URI	Uniform Resource Identifier[RFC1630]	统一资源标识符
URL	Uniform Resource Locator	统一资源定位器
W3C	World Wide Web Consortium	万维网联盟
WAE	Wireless Application Environment[WAE]	无线应用环境
WAP	Wireless Application Protocol[WAP]	无线应用协议
WTA	Wireless Telephony Applications[WTA]	无线电话应用
WTAI	Wireless Telephony Applications Interface[WTAI]	无线电话应用接口
WWW	World Wide Web	万维网

9.8 参考标准

- [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>
- [WAE] "Wireless Application Environment Specification" , WAP Forum, 1998
URL: <http://www.wapforum.org/>
- [WAP] "Wireless Application Protocol Architecture Specification, version 0.9" , WAP Forum, 1997
URL: <http://www.wapforum.org/>
- [WML] "Wireless Markup Language" , WAP Forum, 1998
URL: <http://www.wapforum.org/>
- [WMLScript] "WMLScript Language Specification" , WAP Forum, 1998
URL: <http://www.wapforum.org/>
- [WSP] "Wireless Session Protocol Specification" , WAP Forum, 1998
URL: <http://www.wapforum.org/>
- [WTA] "Wireless Telephony Application Specification" , WAP Forum, 1998
URL: <http://www.wapforum.org/>
- [XML] "Extensible Markup Language (XML), W3C Proposed Recommendation 10-February-1998, REC-xml-19980210" , T. Bray, et al., February 10, 1998
URL: <http://www.w3.org/TR/REC-xml>

9.9 参考资料

[RFC1738] “Uniform Resource Locators (URL)”, T. Berners-Lee, et al., December 1994
URL: <http://ds.internic.net/rfc/rfc1738.txt> described in [RFC2119].

9.10 WTA的 URI和 WMLScript函数库

表9-5至表9-9总结了URI和 WMLScript函数库的调用。为了增加可读性，省去了参数部分。
“库/函数 ID”一栏中的数字，表示函数库和函数的标识符。

9.10.1 公共的WTA

公共的 WTAI库（见表9-5）

表9-5 公共的WTAI库

库/函数ID	URI	脚本调用	描述
512.0	Wtai://wp/mc	WTAPublic.makeCall	建立由移动端发起的呼叫

9.10.2 网络公用WTA

1. 呼叫控制库（见表9-6）

表9-6 呼叫控制库

库/函数ID	URI	脚本调用	描述
513.0	wtai://cc/sc	WTACallCont.setup	建立一个新呼叫
513.1	wtai://cc/ac	WTACallCont.accept	接受来话呼叫
513.2	wtai://cc/rc	WTACallCont.release	释放呼叫
513.3	wtai://cc/sd	WTACallCont.sendDTMF	发送DTMF拨号音

2. 网络文本库（见表9-7）

表9-7 网络文本库

库/函数ID	URI	脚本调用	描述
514.0	—	WTANetText.send	发送网络文本
514.1	—	WTANetText.read	读网络文本
514.2	—	WTANetText.delete	删除网络文本
514.3	—	WTANetText.getFieldValue	获得字段值

3. 电话簿库（见表9-8）

表9-8 电话簿库

库/函数ID	URI	脚本调用	描述
515.0	—	WTAPhoneBook.write	写电话簿记录
515.1	—	WTAPhoneBook.read	读电话簿记录
515.2	—	WTAPhoneBook.delete	删除电话簿记录
515.3	—	WTAPhoneBook.getFieldValue	获得字段值

4. 其他库（见表9-9）

表9-9 其他库

库/函数ID	URI	脚本调用	描述
516.1	—	WTAMisc.indication	逻辑指示
516.2	wtai://ms/ec	WTAMisc.endcontext	结束用户代理上下文

9.11 WTAI预先确定的错误代码

WTA函数库中的函数可以返回一个结果代码，指出函数调用的最终结果（见表 9-10），在大多数情况下，正整数表示调用成功。WTAI定义了可以被 WTAI函数返回的一个错误代码（也就是非正的结果代码）的集合。需要注意的是：并不是所有的函数都要使用全部的错误代码。代码- 1到- 63被保留，用于支持 WTA的标准库函数，特殊网络的 WTA必须使用从- 64到- 127的代码。

表9-10 WTAI定义误码

错误代码	描述
-1	ID没有找到，函数运算不能完成
-2	不合法的参数。由于丢失了参数，函数无法求解
-3	无法使用的服务或者不存在这个功能
-4	服务暂时无法使用
-5	被叫方占线
-6	网络占线
-7	没有应答，也就是呼叫建立过程超时
-8 ~ -63	保留，用于将来的WTA标准库函数
-64 ~ -127	特殊网络的错误代码

9.12 WTAI应用举例

WTAI函数可以用两种方式调用：第一，用 URL方案完成调用；第二，通过一个脚本，实现调用。下面的两个例子演示了如何利用 WML和WMLScript解决一个简单的问题。

用URL方案调用 WTAI函数的例子如下：

```
<WML>
<CARD>
  <DO TYPE="ACCEPT" TASK="GO" URL="#eFood" />
  Welcome!
</CARD>
<CARD NAME="eFood">
  <DO TYPE="ACCEPT" TASK="GO" URL="wtai://cc/mc;$FoodNum" />
  Choose Food:
  <SELECT KEY="FoodNum">
    <OPTION VALUE="5556789">Pizza</OPTION>
    <OPTION VALUE="5551234">Chinese</OPTION>
    <OPTION VALUE="5553344">Sandwich</OPTION>
    <OPTION VALUE="5551122">Burger</OPTION>
```

```

    <SELECT>
  </CARD>
</WML>

```

通过脚本调用 WTAI函数的例子如下：

WMLSCRIPT:

```

function CallFood(N) {
  var i = wtaCallControl.Setup(N;1);
  if (i == 0) {
    // Call is good, show call is done
    Browser.setVar("Msg"      , "Called");
    Browser.setVar("Nmbr"     , N);
  }
  else {
    // Call failed, we could tell user why
    Browser.setVar("Msg"      , "Error");
    Browser.setVar("Nmbr"     , $i);
  }
  Browser.go("displayMsg");
}

```

<WML>

```

<CARD>
  <DOCTYPE="ACCEPT" TASK="GO" URL="/script#CallFood($FoodNum)" />
  Choose Food:
  <SELECT KEY="FoodNum">
    <OPTION VALUE="5556789">Pizza</OPTION>
    <OPTION VALUE="5551234">Chinese</OPTION>
    <OPTION VALUE="5553344">Sandwich</OPTION>
    <OPTION VALUE="5551122">Burger</OPTION>
  <SELECT>
</CARD>
<CARD NAME="displayMsg">
  Call Status: $Msg $Nmbr
</CARD>
</WML>

```