

文章编号:1004 - 5694(2001)02 - 0062 - 05

WAP 与 WWW 网页编程比较*

蹇洁, 陈前斌

(重庆重邮信科股份有限公司, 重庆 400065)

摘要:对 WAP 网页编程与传统的 WWW 编程在编程模型、服务器设置、编程语法等方面作了详细比较, 并阐明了 WAP 网页编程的方法。

关键词:无线应用协议;编程模型;网页编程

中图分类号:TP393 **文献标识码:**A

The Comparison of WAP Page Programming with Web Page Programming

JIAN Jie, CHEN Qian-bin

(Chongqing Chongyou Information Technology Co. Ltd., Chongqing University of Posts and Telecommunications, Chongqing 400065, China)

Abstract: In this paper, the authors compare WAP programming with the traditional WWW programming in terms of the model of programming, the set of server and the grammar of programming. The order of this paper is to help readers to comprehend and understand WAP program.

Key words: WAP; programimg model; page programming

1 编程模型对比

WWW(World Wild Web)万维网,简称 Web,它是由计算机通过电话线或直接线路链接进行网点和网页对话的网络,其编程模型如图 1 所示。在使用 Web 浏览器的 WWW 模型中,Web 浏览器分析用户请求的 URL(统一资源定位器),并向 Web 服务器发送一个 HTTP 或 HTTPS(安全超文本传输协议)请求;Web 服务器接收并分析请求后,将相应内容以 HTTP 形式回送。

WAP(Wireless Application Protocol)无线应用协议是针对移动终端通过各种无线网络接入 Internet 的无线应用层协议标准。WAP 编程模型是

以 WWW 编程模型为基础的,如图 2 所示。客户端

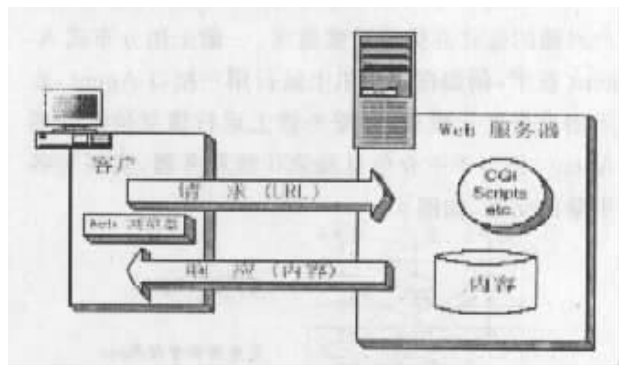


图 1 WWW 编程模型

通过用户代理(如手机的微型浏览器)发出一个 WAP 请求, WAP 网关将 WAP 请求转换成 WWW 请求,并提交给目标内容服务器,再将目标内容服务

* 收稿日期:2000 - 12 - 01

作者简介:蹇洁(1973-),女,主要研究方向为移动通信技术;陈前斌,男,副教授,硕士生导师,主要研究方向为移动通信技术。

器返回的, 用 HTML 语言标识的响应翻译成 WML 语言标识的响应, 编码成二进制格式回送给终端。

WAP 编程模型与 WWW 编程模型相似, 最突出的差异有 2 点。

- 在用户代理和内容服务器之间总是至少有一个 WAP 专用代理服务器。该代理服务器的主要任务是将来自用户代理的 WAP 协议翻译成 HTTP, 以便与内容服务器通信, 反之亦然。

- 用户代理和 WAP 代理服务器间的通信由 WAP 协议完成。这些协议中最重要的是无线会话协议(WSP), 它将 WAP 的内容转化为紧凑的二进制编码格式以减少网络的业务量。

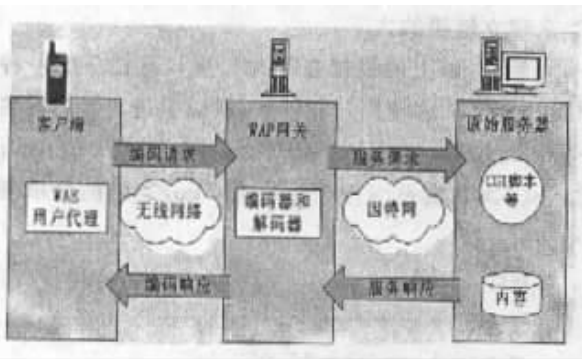


图 2 WAP 编程模型

2 WAP 服务器设置

同设计 WWW 时相同, 在设计 WAP 网页时也需要进行 Web 服务器设置:

(1) 对于 IIS4, 可在其管理器里的站点属性中加入新的文件类型, 后缀为 .wml 和 MIME 类型 text/vnd.wap.wml;

(2) 对于 PWS, 可修改注册表, 先在 HKEY_CLASSES_ROOT 层加入主键 .wml, 再加入串值 Content Type 为 text/vnd.wap.wml, 然后在 HKEY_LOCAL_MACHINE\Software\CLASSES\MIME\Database\Content Type\ 中加入主键 text/vnd.wap.wml, 最后加入串值 Extension 为 .wml;

(3) 对于 Apache 可修改 Apache 安装目录下的 conf/mime.types 文件, 该文件中增加以下内容:

```
text/vnd.wap.wml      .wml
image/vnd.wap.wbmp   .wbmp
```

```
application/vnd.wap.wmlc      .wmlc
text/vnd.wap.wmls            .wmls
application/vnd.wap.wmlsc     .wmlsc。
```

3 WAP 网页编程

制作 WWW 网页使用 HTML (Hypertext Markup Language), 即超文本标记语言。制作 WAP 网页使用 WML (Wireless Markup Language), 即无线标记语言, 它是一种类似 HTML 的面向标签语言。WML 文档是由一个卡面(deck)和若干个卡片(card)组成的, 与 HTML 的页面类似, 一张卡片通常包含一些可视的内容, 也会包含一些用户的选择。

3.1 基本规则

HTML 的元素习惯上是用大写字母, 文档的变量包含在 javascript 中。而 WML 的所有标签、属性和规定的可接收值必须小写, 对于不成对出现的标签, 必须在 > 前加 /, 比如
。文档变量可以定义在 card 中。WML 的实体代码与 HTML 相同。

3.2 基本格式和文件头

wml 文件的一般格式:

```
<? xml version="1.0" ? >
<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML
1.1//EN"
"http://www.wapforum.org/DTD/wml-1.1.
xml">
<wml>
  <head>
    . head information
  </head>
  <card>
    . card definition
  </card>
  . optional additional cards
</wml>
```

结构看上去和 HTML 文件很类似, 对于每一个 DECK (即一个 WAP 网页), 在其文档开头的类型声明:

```
<? xml version="1.0" ? >
<! DOCTYPE wml PUBLIC "-//WAPFORUM//DTD
WML 1.1//EN" "http://www.wapforum.org/DTD/wml-
```

1. 1. xml" >是被 WAP 编辑器和服务器用来进行 WAP 程序的创建和执行的。对于一个 DECK,其文件大小最好不要超过 1K.

<wml> 标签和 HTML 中的 <html> 标签一样,用来表明这是一个 WML 的 DECK。

<head>和 HTML 一样,标签包含了该 DESK 的相关信息。<head> 标签之间可以包含一个 <access> 标签和多个 <meta> 标签。

<access .../> 相当于 HTML 中的 <BASE> 标签,指定该 DECK 的访问控制信息。

<meta .../> 和 HTML 中的类似,提供了该 DECK 的 meta 信息。

一个 DECK 可以包含多个 CARD,每个 CARD 显示的内容就像 HTML 的一个网页,用 <card> 和 </card> 包含。

3.3 显示文本

在文本的显示上 WML 基本和 HTML 相同。文字段落包含在 <p...> 和 </p> 之间。

换行标签也一样为
。

文字的修饰标签有 、<i>、<u>、、、<big> 和 <small>,意义和 HTML 里的相同。

表格的显示标签也和 HTML 相近,使用 <table ...>、<tr>和 <td>来显示。

test1.wml

<? xml version="1.0"? >

<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.1//EN"

"http://www.wapforum.org/DTD/wml_1.1.xml">

<wml>

<card>

<p>

<i> Appointments </i>

</p>

<p>

<table title="mytable" columns="2">

<tr><td>Date</td><td>Start/Stop</td></tr>

<tr><td>11/25</td><td>9:00-11:00</td>

<tr><td>12/03</td><td>4:00-6:00</td>

执行结果

| Appointments | |
|--------------|------------|
| Date | Start/Stop |
| 11/25 | 9:00-11:00 |
| 12/03 | 4:00-6:00 |

OK

</tr>

<tr><td>12/03 </td><td>4:00-6:00</td>

</tr>

</table>

</p>

</card>

</wml>

3.4 显示图片

显示图片(1 位黑白 BMP 图片)的标签一样类似于 HTML,.

3.5 锚和任务

链接是 HTML 页面里最基本的功能,在 WML 里也一样用 和 来建立链接的文字。

然而,以上的链接在 WML 里只是任务的一种情况,为了能够使用其它任务,所以引进了新的标签 <anchor title="label">任务标签 文本</anchor>,<a> 其实是当任务标签为 <go/> 时的简单表示方式。

test3.wml

<? xml version="1.0"? >

<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.1//EN"

"http://www.wapforum.org/DTD/wml_1.1.xml">

<wml>

<card>

<p>

<anchor title="Link1"><go href="

test1.wml"/>News</anchor>

Sports

</p>

</card>

</wml>

执行结果:

| |
|---------|
| News |
| Sports |
| options |

WML 里的任务标签有以下几种,除了用于 <anchor>,还可以用在下面事件中。

• <go> 用来指示浏览器显示指定的 URL,包括 DECK 的第一个 CARD 或者指定的 CARD,语法为:

<go href="url" .../>.

• <prev> 用来将当前页面的 URL 压入 URL 的历史堆栈,并打开此前的 URL。语法类似为

<go>.

- <refresh>用来刷新当前的页面,从而使得页面内的变量刷新。

- <noop>表示什么也不做。

3.6 显示表单

类似于 HTML 的<FORM>, <fieldset> 用来包括一组表单选项。语法为<fieldset title="label">表单内容</fieldset>。表单内容可以是嵌套的<fieldset>, 输入框<input>, 选单<select> 和必要的提示文本。

WML 也用<input name="variable" .../>, 用来输入文本。

```
Test3.wml
-----
<? xml version="1.0"? >
<! DOCTYPE wml PUBLIC "-//WAPFORUM//DTD
WML 1.1//EN"
```

```
"http://www.wapforum.org/DTD/wml-1.1.xml">
<wml>
<card >
<p>
  USERNAME:
    <input name="userName" />
  PASSWORD:
    <input name="password" type="password" />
  COMMENTS:
    <input name="comments" />
</p>
</card>
</wml>
```

执行结果

| |
|------------------------------------|
| USERNAME: <input type="text"/> |
| PASSWORD: <input type="password"/> |
| COMMENTS: <input type="text"/> |
| option |

<select> 选单类似于 HTML 表单中的<SELECT>, <select> 和</select> 间可包含<option> 标签, 语法如下:

```
<select ...>
  <option title="label" onpick="url" ...>
    事件或者文本
  </option>
</select>
```

<option>, 类似于 HTML 中选单的<OPTION>, 用来表示选单的可选项。<option> 和</option> 间可包括事件和选单的显示文本。

onpick 属性, 用来指定用户选到该项并按 ACCEPT 键后所打开页面的 URL。

例见 test5.wml

3.7 事件

WML 的事件基本上分为 2 大类, 一类是键盘 (包括软硬按钮) 输入事件, 用<do> 标签来处理; 另一类是相关页面内部的事件, 用<onevent> 标签来处理。

<do> 的语法如下: <do type="type"...> 任务</do>, 任务就是以前提到的 4 种任务。

* type 属性指定触发的事件, 具体如下:

| | |
|---------|-----------------|
| type 值 | 触发原因: |
| accept | 调用 ACCEPT 按钮机制; |
| delete | 调用 DELETE 按钮机制; |
| help | 调用 HELP 按钮机制; |
| options | 调用选择按钮机制; |
| prev | 调用 PREV 按钮机制。 |

```
test5.wml
-----
<? xml version="1.0"? >
<! DOCTYPE wml PUBLIC "-//WAPFORUM//DTD
WML 1.1//EN"
```

```
"http://www.wapforum.org/DTD/wml-1.1.xml">
<wml>
<head>
  <meta http-equiv="Cache-Control" content="
max-age=0"/>
</head>
<card id="card0" ordered="false">
  <do type="accept" label="InputName" name="
do1">
    <go href="#card01"/>
  </do>
  <p>
    NAME:<input name="userName" />
  </p>
</card>
<card id="card01">
  <p>
    You name is $(userName.noesc).
  </p>
</card>
```

```
</wml>
```

<onevent> 的语法如下: <onevent type = "type">任务</onevent>, 属性 type 的取值如下: type 值 如用户执行了以下操作就执行任务; onpick 用户选择或不选一个<option>项时; onenterforward 用户使用<go>任务到达一个 CARD 时;

onenterbackward 用户使用<prev>任务返回到前面的 CARD 时, 或者按 BACK 按钮时;

ontimer 当<timer>过期时。

```
Test5.wml
```

```
-----
```

```
<? xml version="1.0"? >
```

```
<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.1//EN"
```

```
"http://www.wapforum.org/DTD/wml_1.1.xml">
```

```
<wml>
```

```
<head>
```

```
<meta http-equiv="Cache-Control" content="max-age=0"/>
```

```
</head>
```

```
<card id="card0" ordered="false">
```

```
<p>
```

```
Please select a city...
```

```
<select title="Cities List" name="city">
```

```
<option title="Beijing" value="Beijing">
```

```
<onevent type="onpick">
```

```
<go href="#card01"/>
```

```
</onevent>Beijing</option>
```

```
<option title="Shanghai" value="Shanghai" onpick=" #card01">Shanghai</option>
```

```
<option title="Hongkong" value="Hongkong" onpick=" #card01">Hongkong</option>
```

```
</select>
```

```
</p>
```

```
</card>
```

```
<card id="card01">
```

```
<p>
```

```
You are Living in $(city:noesc)
```

```
</p>
```

```
</card>
```

```
</wml>
```

3.8 CGI 编程

和普通 CGI 编程的方式差不多, 只是返回结果时, 先要输出 Content-Type 为 text/vnd.wap.wml, 然后再输出 WML 内容。

WAP 编程模型是以 WWW 编程模型为基础的, 面向窄带传输, 适合微浏览器。