

## UNIT III :

Variables – Other Content you can Include – Controls – Miscellaneous Markup – Sending Information – Application Security – Other Data: The Meta Element – Document Type-Declarations – Errors and Browser Limitations – Content Generation – WML Version Negotiation.

### UNIT III

#### Variables:

Variables allow you to alter your content dynamically at run time. Variables are considered set if they contain any value other than the empty string.

The syntax of accessing the value of a variable is the U.S. dollar sign followed by the variable's name in parentheses: \$(user name). If the variable reference is followed by whitespace, you can omit the parentheses:

**Your name is \$first \$(last).**

Variables can be created and set using several different methods. Following are two examples:

#### The <setvar> element:

The <setvar> element is used as a result of the user executing some task. The >setvar> element can be used to set a variable's state within the following elements: <go>, <prev>, and <refresh>.

This element supports the following attributes:

Attribute	Value	Description
name	string	Sets the name of the variable
value	string	Sets the value of the variable
class	class data	Sets a class name for the element.
id	element ID	A unique ID for the element.

The following element would create a variable named *a* with a value of 1000:

```
<setvar name="a" value="1000"/>
```

#### The input elements:

Variables are also set through any input element like *input*, *select*, *option*, etc. A variable is automatically created that corresponds with the named attribute of an input element.

For example, the following element would create a variable named *b*:

```
<select name="b">  
<option value="value1">Option 1</option>  
<option value="value2">Option 2</option>  
</select>
```

Using Variables:

Variable expansion occurs at runtime, in the microbrowser or emulator. This means it can be concatenated with or embedded in other text.

Variables are referenced with a preceding dollar sign, and any single dollar sign in your WML deck is interpreted as a variable reference.

### Initial Variable

```
<setvar name="var1" value=""/>
```

### Other content you can include

WML documents may include several other types of content, including images, tables, and links.

### Images:

As in HTML, decks cannot images directly; instead, they can include references to images, which exists in separate files.

A picture may be worth a thousand words, but no matter how they are presented, no one wants to read a thousand words on a four-line screen.

The `<img>` element is used to include an image in a WAP card. WAP-enabled wireless devices only supported the Wireless Bitmap (WBMP) image format.

WBMP images can only contain two colors: black and white. The file extension of WBMP is ".wbmp" and the MIME type of WBMP is "image/vnd.wap.wbmp".

The `<img>` element supports the following attributes:

Attribute	Value	Description
align	<ul style="list-style-type: none"><li>• top</li><li>• middle</li></ul>	Alignment of the image

	<ul style="list-style-type: none"> <li>• bottom</li> </ul>	
alt	alternative text	Sets an alternate text to be displayed if the image is not displayed.
height	<ul style="list-style-type: none"> <li>• px</li> <li>• %</li> </ul>	Height of the image in pixels or percentage. If you specify the value in pixels, the syntax is "140", instead of "140px".
hspace	<ul style="list-style-type: none"> <li>• px</li> <li>• %</li> </ul>	Sets white space to the left and right of the image. If you specify the value in pixels, the syntax is "140", instead of "140px".
localsrc	cdata	Sets an alternate representation for the image. If this attribute is set, the browser will use it instead of the "src" attribute.
src	image url	A path to wbmp image.
vspace	<ul style="list-style-type: none"> <li>• px</li> <li>• %</li> </ul>	Sets white space above and below the image. If you specify the value in pixels, the syntax is "140", instead of "140px".
width	<ul style="list-style-type: none"> <li>• px</li> <li>• %</li> </ul>	Sets the width of the image. If you specify the value in pixels, the syntax is "140", instead of "140px".
xml:lang	language_code	Sets the language used in the element
class	class data	Sets a class name for the element.
id	element ID	A unique ID for the element.

Following is the example showing usage of <img> element.

```
<?xml version="1.0"?>
<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.2//EN"
"http://www.wapforum.org/DTD/wml12.dtd">

<wml>

<card title="WML Images">
<p>
```

```

This is Thumb image

</p>

<p>
This is Heart image

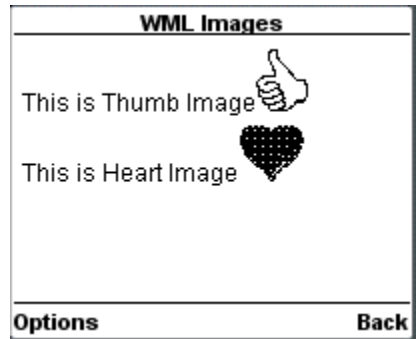
</p>

</card>

</wml>

```

This will produce the following result:



### T Tables:

WML support simple tables for organizing content into columns. These tables are not intended to provide fancy formatting—bear in mind the small size of the screen that most devices will use.

Table data is organized into rows, specified with the tr (table row) element; rows, in turn, contain cells indicate by td (table data) elements.

The <table> element along with <tr> and <td> is used to create a table in WML. WML does not allow the nesting of tables

A <table> element should be put with-in <p>...</p> elements.

The <table /> element supports the following attributes:

Attribute	Value	Description
-----------	-------	-------------

columns	number	Sets the number of columns in the table
align	<ul style="list-style-type: none"> <li>• L</li> <li>• C</li> <li>• R</li> </ul>	<p>To specify the horizontal text alignment of the columns, you need to assign three letters to the align attribute. Each letter represents the horizontal text alignment of a column. The letter can be L, C, or R. For example, if you want the following settings to be applied to your table:</p> <ul style="list-style-type: none"> <li>• First table column -- Left-aligned</li> <li>• Second table column -- Center-aligned</li> <li>• Third table column -- Right-aligned</li> </ul> <p>Then you should set the value of the <i>align</i> attribute to LCR.</p>
xml:lang	language_code	Sets the language used in the element
class	class data	Sets a class name for the element.
id	element ID	A unique ID for the element.

Following is the example showing usage of <table> element.

```

<?xml version="1.0"?>
<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.2//EN"
"http://www.wapforum.org/DTD/wml12.dtd">

<wml>

<card title="WML Tables">
<p>
<table columns="3" align="LCR">
    <tr>
        <td>Col 1</td>
        <td>Col 2</td>
        <td>Col 3</td>
    </tr>

```

```
<tr>
  <td>A</td>
  <td>B</td>
  <td>C</td>
</tr>

<tr>
  <td>D</td>
  <td>E</td>
  <td>F</td>
</tr>
</table>
</p>
</card>

</wml>
```

This will produce the following result:

WML Tables		
Col 1	Col 2	Col 3
A	B	C
D	E	F
<b>Options</b>		<b>Back</b>

WML – Links

### Links:

HTML standardized an unfortunate overloading of the term anchor, applying it not only to the point in the document that activates the link.

A WML link is a connection between a document hotspot and a task. This task might be a navigation task (mimicking HTML links) but could also be a prev or refresh task.

WML provides you an option to link various cards using links and then navigate through different cards.

There are two WML elements, <anchor> and <a>, which can be used to create WML links.

WML <anchor> Element:

The <anchor>...</anchor> tag pair is used to create an anchor link. It is used together with other WML elements called <go/>, <refresh> or <prev/>. These elements are called task elements and tell WAP browsers what to do when a user selects the anchor link

You can enclose Text or image along with a task tag inside <anchor>...</anchor> tag pair.

The <anchor> element supports the following attributes:

Attribute	Value	Description
title	cdata	Defines a text identifying the link
xml:lang	language_code	Sets the language used in the element
class	class data	Sets a class name for the element.
id	element ID	A unique ID for the element.

Following is the example showing usage of <anchor> element.

```
<?xml version="1.0"?>
<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.2//EN"
"http://www.wapforum.org/DTD/wml12.dtd">

<wml>

<card title="Anchor Element">
```

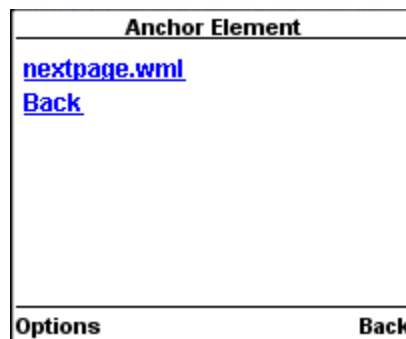
```

<p>
  <anchor>
    <go href="nextpage.wml"/>
  </anchor>
</p>
<p>
  <anchor>
    <prev/>
  </anchor>
</p>
</card>

</wml>

```

This will produce the following result:



WML <a> Element:

The <a>...</a> tag pair can also be used to create an anchor link and always a preferred way of creating links.

You can enclose Text or image inside <a>...</a> tags.

The <a> element supports the following attributes:

Attribute	Value	Description
href	URL	Defines URL of the liked page



title	cdata	Defines a text identifying the link
xml:lang	language_code	Sets the language used in the element
class	class data	Sets a class name for the element.
id	element ID	A unique ID for the element.

Following is the example showing usage of <a> element.

```

<?xml version="1.0"?>
<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.2//EN"
"http://www.wapforum.org/DTD/wml12.dtd">

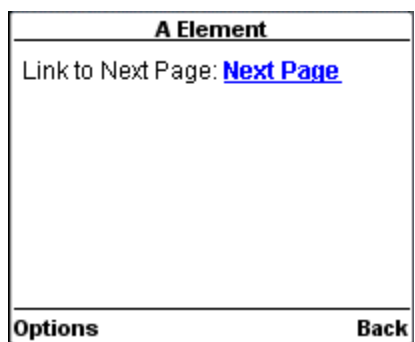
<wml>

<card title="A Element">
<p> Link to Next Page:
  <a href="nextpage.wml">Next Page</a>
</p>
</card>

</wml>

```

This will produce the following result:



## WML <a>Tag

The <a>...</a> tag pair can also be used to create an anchor link and always a preferred way of creating links.

You can enclose Text or image inside <a>...</a> tags.

Attributes:

This element supports the following attributes:

Attribute	Value	Description
href	URL	Defines URL of the liked page
title	cdata	Defines a text identifying the link
xml:lang	language_code	Sets the language used in the element
class	class data	Sets a class name for the element.
id	element ID	A unique ID for the element.

Example:

Following is the example showing usage of this element.

```
<?xml version="1.0"?>
<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.2//EN"
"http://www.wapforum.org/DTD/wml12.dtd">

<wml>

<card title="A Element">
<p> Link to Next Page:
  <a href="nextpage.wml">Next Page</a>
</p>
```

</card>

</wml>

This will produce the following result:

<b>A Element</b>	
Link to Next Page: <a href="#">Next Page</a>	
<b>Options</b>	<b>Back</b>

## Controls:

Bindings and links, WML provides two elements that allow the user to provide data to your application: the input element defines entry fields, and the select element defines menus.

The <input/> element is used to create input fields and input fields are used to obtain alphanumeric data from users.

Attributes:

This element supports the following attributes:

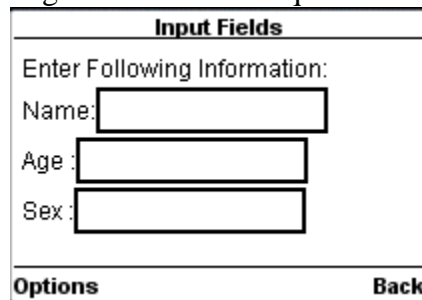
Attribute	Value	Description
name	text	The name of the variable that is set with the result of the user's input
maxlength	number	Sets the maximum number of characters the user can enter in the field
emptyok	<ul style="list-style-type: none"><li>• true</li><li>• false</li></ul>	Sets whether the user can leave the input field blank or not. Default is "false"
format	A a N X x M m *f nf	Sets the data format for the input field. Default is "*M". A = uppercase alphabetic or punctuation characters a = lowercase alphabetic or punctuation characters N = numeric characters X = uppercase characters x = lowercase characters M = all characters m = all characters *f = Any number of characters. Replace the <i>f</i> with one of the letters above to specify what characters the user can enter nf = Replace the <i>n</i> with a number from 1 to 9 to specify the number of characters the user can enter. Replace the <i>f</i> with one of the letters above to specify what characters the user can enter
size	number	Sets the width of the input field
tabindex	number	Sets the tabbing position for the select element
title	text	Sets a title for the list
type	<ul style="list-style-type: none"><li>• text</li><li>• password</li></ul>	Indicates the type of the input field. The default value is "text". Password field is used to take password for authentication purpose.
value	text	Sets the default value of the variable in the "name" attribute
xml:lang	language_code	Sets the language used in the element
class	class data	Sets a class name for the element.
id	element ID	A unique ID for the element.

Example:

Following is the example showing usage of this element.

```
<?xml version="1.0"?>
<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.2//EN"
"http://www.wapforum.org/DTD/wml12.dtd">
<wml>
<card title="Input Fields">
<p> Enter Following Information:<br/>
Name: <input name="name" size="12"/>
Age : <input name="age" size="12" format="*N"/>
Sex : <input name="sex" size="12"/>
</p>
</card></wml>
```

This will provide you the following screen to enter required information:



### WML <select> Tag

The <select>...</select> WML elements are used to define a selection list and the <option>...</option> tags are used to define an item in a selection list. Items are presented as radio buttons in some WAP browsers. The <option>...</option> tag pair should be enclosed within the <select>...</select> tags.

#### Attributes:

This element supports the following attributes:

Attribute	Value	Description
iname	text	Names the variable that is set with the index result of the selection
ivalue	text	Sets the pre-selected option element
multiple	<ul style="list-style-type: none"><li>• true</li><li>• false</li></ul>	Sets whether multiple items can be selected. Default is "false"
name	text	Names the variable that is set with the result of the selection
tabindex	number	Sets the tabbing position for the select element
title	text	Sets a title for the list
value	text	Sets the default value of the variable in the "name" attribute

xml:lang	language_code	Sets the language used in the element
class	class data	Sets a class name for the element.
id	element ID	A unique ID for the element.

Example:

Following is the example showing usage of this element.

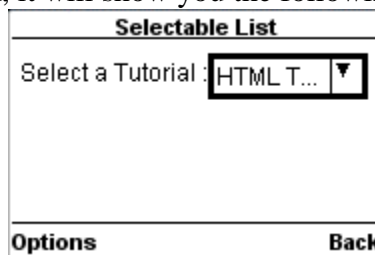
```
<?xml version="1.0"?>
<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.2//EN"
"http://www.wapforum.org/DTD/wml12.dtd">

<wml>

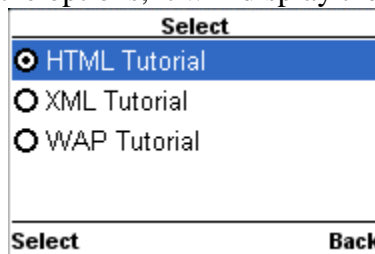
<card title="Selectable List">
<p> Select a Tutorial :
<select>
<option value="htm">HTML Tutorial</option>
<option value="xml">XML Tutorial</option>
<option value="wap">WAP Tutorial</option>
</select>
</p>
</card>

</wml>
```

When you will load this program, it will show you the following screen:



Once you highlight and enter on the options, it will display the following screen:



You want to provide option to select multiple options, then set *multiple* attribute to *true* as follows:

```
<?xml version="1.0"?>
<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.2//EN"
"http://www.wapforum.org/DTD/wml12.dtd">

<wml>
```

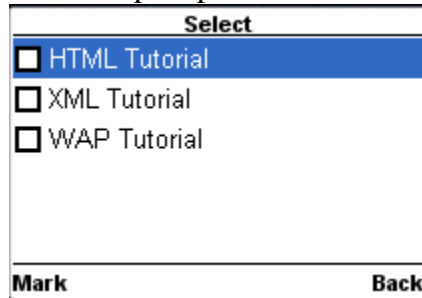
```

<card title="Selectable List">
<p> Select a Tutorial :
<select multiple="true">
  <option value="htm">HTML Tutorial</option>
  <option value="xml">XML Tutorial</option>
  <option value="wap">WAP Tutorial</option>
</select>
</p>
</card>

</wml>

```

This will give you a screen to select multiple options as follows:



### Miscellaneous Markup:

The timer element lets you trigger an event after a certain amount of time has elapsed. The fielded element lets you group fields into logical collection.

#### WML <timer> Tag

A timer is declared inside a WML card with the <timer> element. It must follow the <onevent> elements if they are present. (If there are no <onevent> elements, the <timer> must be the first element inside the <card>.) No more than one <timer> may be present in a card.

Attributes:

This element supports the following attributes:

Attribute	Value	Description
name	text	Sets a name for the element.
value	number	Specifies the timer after which timer will be expired. Timeouts are specified in units of a tenth of a second.
class	class_data	Sets a class name for the element.
id	element ID	A unique ID for the element.

Example:

Following is the example showing usage of <timer> element.

```

<?xml version="1.0"?>
<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.2//EN"
"http://www.wapforum.org/DTD/wml12.dtd">

<wml>

```

```

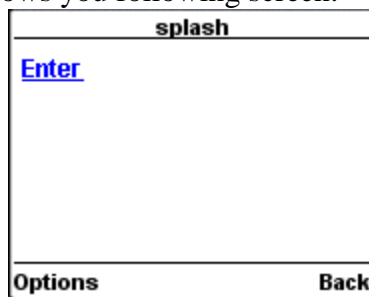
<card id="splash" title="splash">
  <onevent type="ontimer">
    <go href="#welcome"/>
  </onevent>
  <timer value="50"/>
  <p>
    <a href="#welcome">Enter</a>
  </p>
</card>

<card id="welcome" title="Welcome">
  <p>
    Welcome to the main screen.
  </p>
</card>

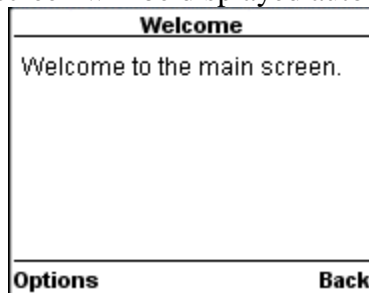
</wml>

```

When you load this program it shows you following screen:



If you do not select given **Enter** option then after 5 seconds you will be directed to **Welcome** page and following screen will be displayed automatically.



### **Sending information:**

You cannot write very interesting applications without some means of sending the user's input to your application server. "get" and "post".

#### **Get versus Post**

The "get" and "post" methods differ in the number of connection that they make to the server and in the amount of data that can be transferred.

With "get", the browser sends the data as part of the document request by appending the data to the URL that it sends to the server.

#### WML <postfield> Tag

The <postfield> tag is used to post variables values to the server.

Attributes:



This element supports the following attributes:

Attribute	Value	Description
name	string	Sets the name of the variable
value	string	Sets the value of the variable
class	class data	Sets a class name for the element.
id	element ID	A unique ID for the element.

Example:

Following example shows how to submit three fields *name*, *age* and *sex* to the server.

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

<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.2//EN"
"http://www.wapforum.org/DTD/wml12.dtd">

<wml>

<card id="card1" title="WML Form">
<p>
Name: <input name="name" size="12"/>

Sex : <select name="sex">
  <option value="male">Male</option>
  <option value="female">Female</option>
</select>

Age : <input name="age" size="12" format="*N"/>
<anchor>
  <go method="get" href="process.php">
    <postfield name="name" value="$(name)"/>
    <postfield name="age" value="$(age)"/>
    <postfield name="sex" value="$(sex)"/>
  </go>
  Submit Data
</anchor>
</p>
</card>

</wml>
```

When you download above code on your WAP device, it will provide you option to enter three fields *name*, *age* and *sex* and one link *Submit Data*. You will enter three fields and then finally you will select *Submit Data* link to send entered data to the server.

### Application Security:

Data security is as critical in the wireless world as it is on the World Wide Web. For, true security, WAP defines the WTLS protocol to provide encrypted, authenticated communications.

WML and WMLScript provide “application-level” security, allowing you to prevent unauthorized decks from linking to yours.

WAP browsers also support the standard Basic Authentication mechanism used on the Web today.

The <access> element provides a simple form of access control for a deck. This allows a deck to specify that only certain other decks may link to it (these decks are known as referring URLs).

There may be no more than one <access> element in a deck, and it must be the first thing inside the <head> element.

Attributes:

The <access> element supports the following attributes:

Attribute	Value	Description
domain	cdata	Specifies the domain (effectively, the range of servers) from which the referring deck must come.
path	cdata	Specifies the path within the referring URL that must match. (The path is the part of the URL that specifies a file or directory on the server.)
class	cdata	Sets a class name for the element.
id	element ID	A unique ID for the element.

Example:

Following is the example showing usage of this element:

```
Access is limited to other decks in www.tutorialspoint.com
<head>
  <access domain="www.tutorialspoint.com"/>
</head>

Access is limited to other decks in www.tutorialspoint.com/wml
<head>
  <access domain="www.tutorialspoint.com" path="/wml"/>
</head>
```

### Other Data: The meta element

The meta element allows you to specify “meta-data”—data about the deck that is contained within, but is not strictly part of, the content.

One common use is to access browser-specific features, such as controlling whether a particular deck can be bookmarked.

### WML <meta> Tag

The <meta> element places an item of arbitrary meta-information in a WML deck. This item is structured as a property name and its value.

You can put any number of <meta> elements into the <head> element. This can add keywords for indexing purposes, store hints about the content of the deck, and store any other information.

Attributes:

The <meta> element supports the following attributes:

Attribute	Value	Description
name	string	Gives the name of this property. Meta-information with this attribute is intended for server-side applications, so it may be removed before it gets to the browser. Could be "keywords", "author", etc.
http-equiv	string	An alternative for the name attribute.
forua	<ul style="list-style-type: none"><li>• true</li><li>• false</li></ul>	If present and set to true, indicates that the property is intended for the use of the browser.
content	string	Should specify a description of the name attribute
scheme	string	Can specify a format or structure that some properties may need to interpret their values. This attribute is used by few properties.
class	class data	Sets a class name for the element.
id	element ID	A unique ID for the element.

Example:

Following is the example showing usage of this element:

```
<?xml version="1.0"?>
<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.2//EN"
"http://www.wapforum.org/DTD/wml12.dtd">

<head>
  <meta name="keyword" content="WML"/>
  <meta http-equiv="Cache-control" content="no-cache"/>
</head>

<wml>

<card id="one" title="First Card">
<p>
This is the first card in the deck
</p>
</card>

<card id="two" title="Second Card">
<p>
Ths is the second card in the deck
</p>
</card>
</wml>
```

## Document Type Declarations

We looked at a skeleton of a WML document, we touched briefly in the Document Type Definition.

The DTD describes the basic syntax of an XML-compliant markup language.

To be valid, every WML deck must contain a Document Type Declaration that references a WML DTD.

### WAP Program Structure

A WML program is typically divided into two parts – the **document prolog** and the **body**. Consider the following code –

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

<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.2//EN"
"http://www.wapforum.org/DTD/wml12.dtd">

<wml>

  <card>

    ...

  </card>

  ...more cards...

</wml>
```

The first line of this text says that this is an XML document and the version is 1.0. The second line selects the document type and gives the URL of the **document type definition (DTD)**. The DOCTYPE declaration specifies the name of the DTD (Document Type Definition) and the URL to the DTD. The DTD contains information about the syntax of the markup language. It defines what elements and attributes can be used in the markup and the rules that they should be used.

It should be placed between the XML declaration and the <wml> element. Below is the DOCTYPE declaration for WML 1.

```
<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.3//EN"
"http://www.wapforum.org/DTD/wml13.dtd">
```

## Errors and Browser Limitations

WML devices tend to have limited resources, users are much more likely to encounter restrictions imposed by device limitations than they are with a conventional browser.

### Content Generation

The browser requests some WML content, the deck that arrives may be static or dynamic.

Dynamic content also includes content generated from templates, where a basic deck reside in the file system and has areas to be “filled in” in response to a particular request.

### **WML Version Negotiation**

The fact that there are numerous versions of the WML standard is not a big issue as this book is published, but it may well become one in the very near future.

WML Versions:

WAP Forum has released a latest version WAP 2.0. The markup language defined in WAP 2.0 is XHTML Mobile Profile (MP). The WML MP is a subset of the XHTML. A style sheet called WCSS (WAP CSS) has been introduced alongwith XHTML MP. The WCSS is a subset of the CSS2.

Most of the new mobile phone models released are WAP 2.0-enabled. Because WAP 2.0 is backward compatible to WAP 1.x, WAP 2.0-enabled mobile devices can display both XHTML MP and WML documents.

WML 1.x is an earlier technology. However, that does not mean it is of no use, since a lot of wireless devices that only supports WML 1.x are still being used. Latest version of WML is 2.0 and it is created for backward compatibility purposes. So WAP site developers need not to worry about WML 2.0.