NOKIA

WAP Service Developer's Guide for the Nokia 9210 Communicator

December 2000

Disclaimer:
Nokia Mobile Phones Limited disclaims all liability, including liability for infringement of any proprietary rights, relating to the implementation of information presented in this document. Nokia Mobile Phones Limited does not warrant or represent that such use will not infringe such rights.
Nokia Mobile Phones Limited retains the right to make changes to this specification at any time without notice.

License:
A license is hereby granted to download and print a copy of this specification for personal use only. No other license to any other intellectual property rights is granted herein.

Contents

1. INTRODUCTION

This Service Developer's Guide for the Nokia 9210 Communicator provides information and practical examples for developers who want to develop WAP services for the Nokia 9210 Communicator. The communicator is an advanced communications device with a feature-rich WAP services application. This document highlights the possibilities, limitations, and requirements of service development and provides a good introduction to the Nokia style of designing user interfaces.

The Wireless Application Protocol (WAP) enhances the functionality of mobile handsets through real-time interactive services. The protocol has been designed especially for small screens and low bandwidth, and it offers a wide variety of wireless services over the Internet for handsets. However, the WAP services screen size in the Nokia 9210 Communicator differs from that of common handsets and should be taken into account when designing services that are both fast and easy to use, and offer the user maximum enjoyability. Besides including issues that are Nokia 9210 Communicator -specific, many of the instructions in this document can be used to maximise interoperability and ease of use of various other browsers.

We at Nokia are very proud of our user interfaces and we have spent a lot of effort to continuously improve them. In this guide, we have collected a lot of what we have learned to help developers improve their services. We believe that good usability will increase use and satisfaction, and thus increase revenues for service providers.

This guide is not intended to describe WAP in detail and it is not meant to replace any WAP specification.

1.1 References

User's guide for the Nokia 9210 Communicator

The Nokia WAP Toolkit can be downloaded from http://www.forum.nokia.com/

Wireless application protocol specifications can be found at http://www.wapforum.org/

Information about Nokia products can be obtained from http://www.nokia.com/

1.2 Contact information

Developer support can be obtained from the Forum Nokia Web site http://www.forum.nokia.com/

Terminology

Card

A single WML unit of navigation and user interface.

Deck

A collection of WML cards that is the smallest downloadable unit. You cannot download a single card in a deck, you must download the entire deck. Service developers should be aware of this, and not create large applications in a single deck.

Hyperlink

A link within a document or card that allows quick navigation to another document (card).

Do element

A way of defining the binding between events (e.g. a user selection) and a task (e.g. to go somewhere).

Input element

A mark-up element that allows for interaction with the user, i.e. letting the user input textual values. These can be used for inputs to locally stored scripts, or for parsing to origin servers.

Proportional fonts

A font in which different letter widths do not necessarily have the same number of pixels. 'W' and 'I' are good examples.

Navigation

The concept of moving between menu items, decks, and cards.

CBA

Command Button Area

PDA

Personal Digital Assistant

CMT

Cellular Mobile Terminal

2. WIRELESS APPLICATION PROTOCOL - WAP

2.1 Introduction

The Wireless Application Protocol (WAP) is a set of protocols that allow the development of applications and services for use with mobile phones and other mobile devices. These protocols and their related standards and specifications are maintained by the WAP Forum.
The WAP Forum consists of a number of high-tech companies from the information technology, software and telecommunications industries. The objectives of the WAP Forum are to:

- Bring Internet content and advanced data applications to digital cellular phones
- Create a global wireless protocol specification that works across different wireless network technologies
- Enable the creation of content and applications that scale across a wide range of bearer networks and device types
- Embrace existing standards and technology wherever possible

2.2 WAP Architecture

2.2.1 WAP and Internet Architecture

In order to leverage on the existing Internet standard as much as possible, the WAP stack closely follows the Internet model. This is illustrated in Figure 1-1.
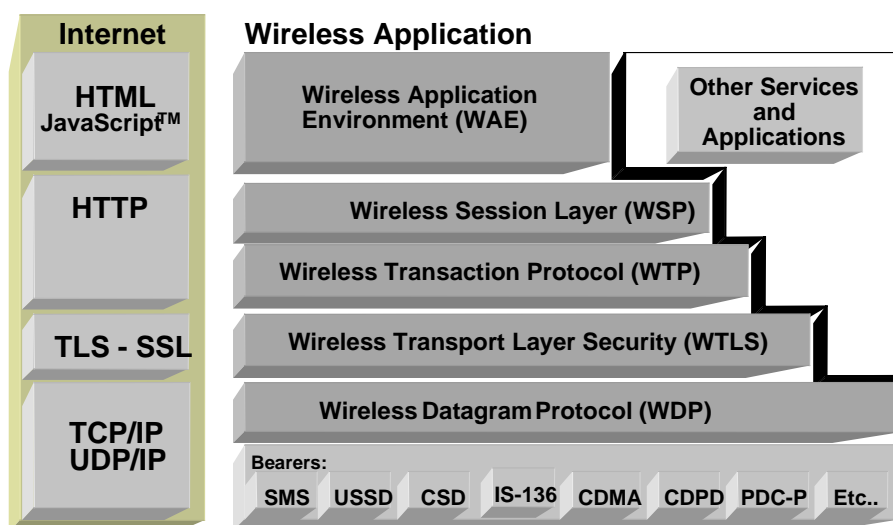

Figure 2-1   Comparison of Internet and WAP Architecture

This layered architecture allows applications to utilise the features of the WAP stack through well-defined interfaces. This close link to the Internet architecture allows developers to utilise their existing knowledge and expertise when developing applications for mobile terminals.

2.2.2 Wireless Application Environment (WAE)

The Wireless Application Environment (WAE) is a general-purpose application environment that uses a combination of Internet and mobile terminal technology. It provides a framework for the development of applications on a mobile terminal.
The WAE contains support for the following functionality:

Wireless Mark-up Language (WML) - a lightweight presentation language, similar to HyperText Mark-up Language (HTML) but optimised for use with mobile terminals.
Wireless Mark-up Language Script (WMLS) - a lightweight script language, similar to Java Script^{TM}.
Wireless Telephony Application / Interface (WTA / WTAI) - telephony services and programming interfaces.
Content formats - defined data formats, such as vCard and vCalendar.

2.2.3 Wireless Session Protocol (WSP)

The Wireless Session Protocol (WSP) provides the application layer of WAP with an interface for two session services. The first is a connection-oriented service that operates above the transaction layer protocol. The second is a connectionless service that operates above a secure or non-secure datagram service.
The WSP is optimised for low bandwidth bearer networks with long latency.

2.2.4 Wireless Transport Protocol (WTP)

The Wireless Transport Protocol (WTP) runs on top of the datagram service and provides a lightweight transaction-oriented protocol suitable for use in mobile terminals. WTP operates over secure or non-secure wireless datagram networks.

2.2.5 Wireless Transport Layer Security (WTLS)

The Wireless Transport Layer Security (WTLS) is based on the industry standard Transport Layer Security (TLS) and is optimised for used over narrow band communication channels. WTLS may be used for secure communication between terminals, and applications can selectively enable WTLS features.

2.3 Developing Applications with WAP

Application developers can use the principles of WAP to develop new services or adapt existing Internet applications for use with mobile terminals. Applications are written in the Wireless Mark-up Language (WML) and the WMLScript, and stored on either a normal Web server (origin server) or directly on the WAP gateway. The content stored on the Web server will be accessible from the mobile terminals via the cellular network and a WAP gateway or proxy.
The proxy server acts as a gateway between the cellular network and the Internet or intranet. The data sent between the origin server and the handset is binary encoded to optimise transmission over the narrow bandwidth of the cellular network. Note that the content stored on the Web server might be in either textual or binary format. When the WAP gateway fetches textual content, it automatically compiles this to the encoded format to minimise network load.
Figure 2-2 shows the network-related elements required for developing and offering services to mobile users.
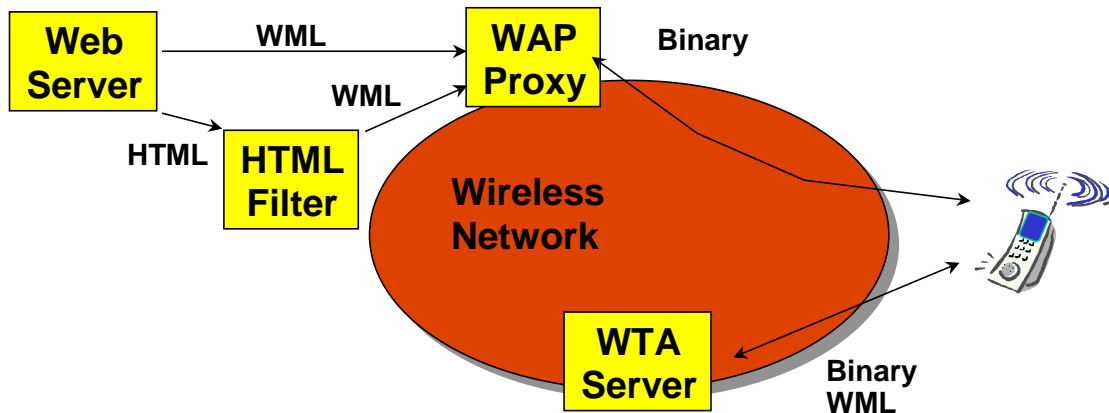
Figure 2-2

Web server
The Web server stores the applications written in WML. Alternatively, existing HTML applications can be used.

HTML filter
Any applications written in HTML will have to be converted to WML before they are sent to the mobile terminal. This HTML filter may form part of the Web server or the WAP proxy.

WAP proxy
The WAP proxy acts as the gateway between the cellular network and the Internet or intranet. It encodes the information, which is to be sent to the mobile terminal in binary format and decodes information sent from the mobile terminal.

WTA server
The WTA server handles network-specific applications. These applications are not discussed in this document.

2.4 Further Information

Further information about WAP is available on the Internet at www.wapforum.org.

3. INTRODUCTION TO THE USER INTERFACE OF THE NOKIA 9210 COMMUNICATOR

Figure 3-1 Communicator interface, WML browser display.

This section gives a short overview of the user interface of the Nokia 9210 Communicator. The Nokia 9210 Communicator is a device with both a communicator interface and a phone interface. The WAP services application is implemented in the communicator interface together with Internet services. The phone interface in the CMT part does not contain a WML browser.

3.1 WAP browser technical features

Data call bearer for WAP.

GSM data transports: HSCSD 2+2 / 3+1 / 1+1 with either 9600 bps or 14400 bps channel coding.

WAP version 1.1

The operating system is EPOC, which is optimised for mobile devices. EPOC is categorised according to screen size. The Nokia 9210 Communicator belongs to the Crystal category (640 x 200 display).

3.2 Keypad

The Nokia 9210 Communicator has a QWERTY keyboard (See figure 3-1) with a menu key for application-specific menus.

3.3 Display – detailed description

3.3.1 Overview

The Nokia 9210 Communicator display (resolution 640x200 pixels) contains the following elements (from left to right): an Indicator area, application area and command button area (CBA) with four soft buttons.

- The width of the indicator area. The width of the wide indicator area is 92 pixels and the width of the narrow indicator area, which is displayed with the WAP browser, is 32 pixels. A space is reserved for different kinds of information, such as inbox and outbox, clock and profile.
- The width of the command button area is dynamic, and depends on the longest command text in the view. The width changes only if the whole view changes, that is, the width remain the same when only the button texts change in the same view. The command texts are aligned to the right. The minimum width of the command button area is 80 pixels, the maximum being 130 pixels.
- The width of the application area is 490 pixels. In full screen mode the width is 625 pixels. Height is always 165 pixels.
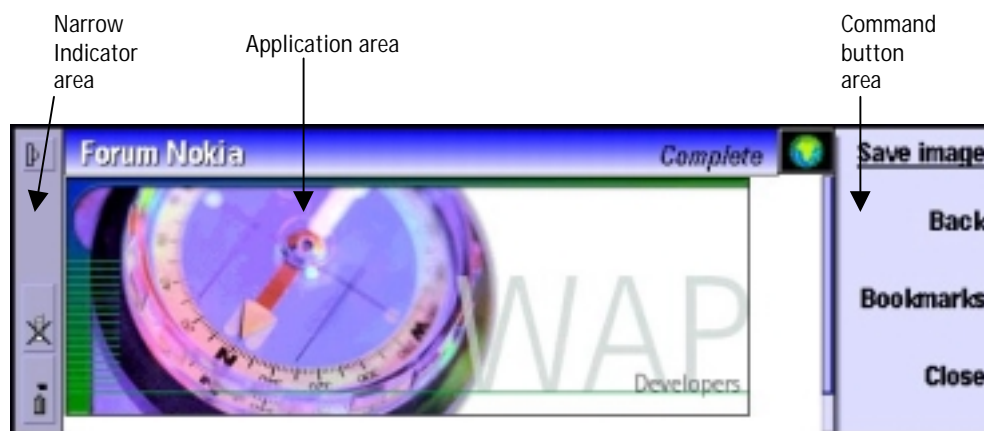
Narrow Indicator area   Application area   Command button area

Figure 3-2   Different parts of the display

3.3.2 Display colormap

Nokia 9210 Communicator supports 4096 colours. The normal "true colour" is specified by 24 bits, 8 bits for each colour component (red, green, and blue). The Nokia 9210 Communicator uses the most significant 4 bits of each of the colour component, resulting in a colour depth of 12 bits. All other colours are either dithered or mapped to the closest colour available, depending on the application.

Although WAP does not support colours, the Nokia 9210 Communicator WAP browser supports image formats with colours (other than the WBMP image format).

3.4 WAP Browser Display

There are two views in the WAP browser: the Bookmarks view and the Card view. The Card view consists of a rendered card, options list, and a title bar. The Bookmarks view consists of a hierarchical list of saved bookmarks and a title bar. Both views have their own menus that provide access to the view's different functions.

Card view

WML cards are displayed in the Card view. Only one card of a WML deck is shown at a time. The vertical scroll bar is visible when the vertical length of the active card exceeds the screen size. The horizontal scroll bar is visible when a table or a non-breaking paragraph exceeds the screen width.

On the left side of the card is an options list, which lists all currently active do element labels. The narrow indicator area and CBA are visible unless in full screen mode. The user can switch focus between the card and the options list with the tab key and the left and right arrow keys.

The browser remembers the focus on both the Card and the options list. For example, if the focus on a card is on an anchor and the user focuses on the options list and then returns the focus to the Card, the focus is automatically set on that same anchor. The same applies to the options list. The browser also remembers whether the focus was on the card or on the options list if the user initiates a task that causes another card to be loaded. Once the new card has been displayed, the focus is set on the card area if it had focus before the action. If the user had focused on the options list before the action, it remains focused after the card has been displayed. The focus on the options list is on the first item after the card has been fetched. The focus is not moved after a refresh task, however.

The first button is a context sensitive 'action' button, that is, when a select or an anchor element is focused, the 1st CBA will be "Select". If no appropriate element has been selected, the button label is dimmed. The first CBA button is always the default button in the card view and may be launched by pressing <ENTER>.

Bookmarks view

In the Bookmarks view, the user creates and manages WAP connections. The Bookmarks view is opened when the browser is launched without an URL. The user can switch to the Bookmarks view from the Card view by selecting the 3rd CBA button, Bookmark. The Bookmarks view consists of a hierarchical list of saved bookmarks, and a title bar.

When the user focuses on a bookmark, its URL will be displayed in an info message. The URL address may be abbreviated, that is, part of the path name may be replaced with three dots, if necessary. The first and the last components of the URL (that is, the domain name and the file name) are visible. If there is no space for both the first and the last component of the URL, the first one is displayed and the URL is simply truncated. If the scheme of the URL is different than "http://", it is displayed before the domain.

Only one bookmark item (a connection bookmark or a folder) can be selected at a time. Zooming is not possible in Bookmarks view.

The first CBA button is always the default button in the Bookmarks view and may be launched by pressing <ENTER>.

4. NOKIA 9210 COMMUNICATOR USER INTERFACE ELEMENTS OF WML

This chapter describes how to use WML in designing services aimed at the Nokia 9210 Communicator. It gives an overview of the graphical user interface elements and their WML capabilities supported by the browser. The document does not include all possible WML elements and attributes. The Nokia 9210 Communicator supports WML v1.1.

The elements are briefly described and shown in sample pictures and WML code. The code examples include only the essential part in using the elements, that is, the document prologue is not shown.
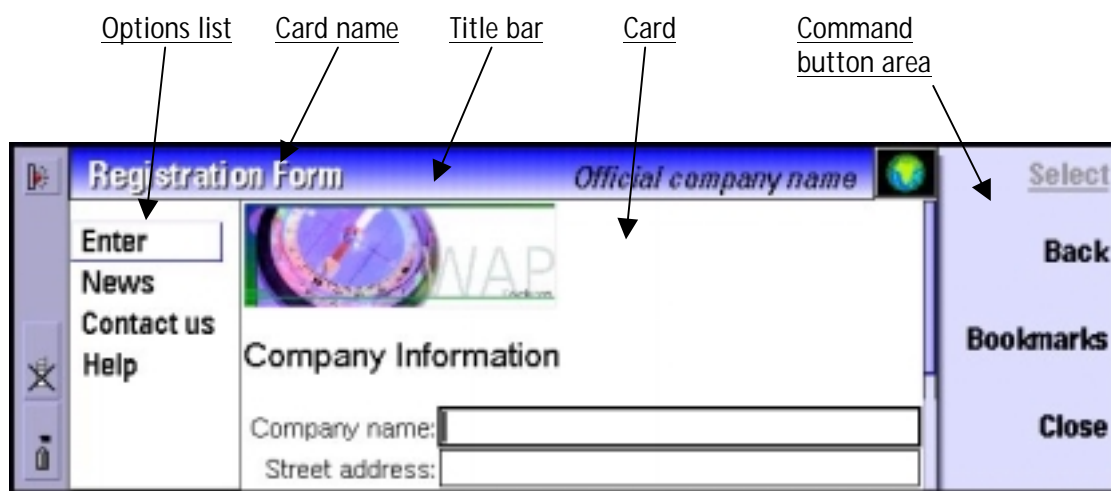
4.1 Browser Display

Figure 4-1   WML browser Card view with different display areas.

The Nokia 9210 Communicator has a high quality active matrix display, with 640 x 200 pixels and 4096 colours. As pixel size is smaller than in laptops, the screen is sharper. In the WAP browser, the application area is divided as follows: 30 pixel title bar on the top, a WML rendering area, and an options list on the left side. The width of options list is 100 pixels, but it is displayed only if there are active do elements on the card. So, depending on the presence of options list, the WML rendering area width is between 390 and 490 pixels. The dimensions in full screen mode are 525 and 625 pixels. The height is always 165 pixels.

The vertical scroll bar is visible when the vertical length of the active card exceeds the screen size. The horizontal scroll bar is visible only when a table or a non-breaking paragraph exceeds the screen width.

On the left side of the card is the options list, which lists all currently active do element labels with their possible icons and hotkeys. The narrow indicator area and CBA are visible unless in full screen mode.

The user can switch focus between the card and the options list with the tab key.

The title bar is divided in two parts. On the left, the title of the card is shown. On the right, the title of the active element is shown. If both titles are too long to fit the title bar, the title of the active element is displayed as whole and the title of the card is truncated or, both of them are truncated. If there is no card name, the URL is displayed instead.

4.2 Formatting elements

The table below contains all text layout-formatting elements. These are described in more details in this chapter later on.

Element	Formatting
em	Italics font of colour red
Strong	Bold, italics font of colour green
b	Bold font
I	Italics font
U	Underlined font
Big	Font size is set to big
Small	Font size is set to small
P	A paragraph break is inserted in the text
Br	A line break is inserted in the text

Table 4-1   Text layout formats.

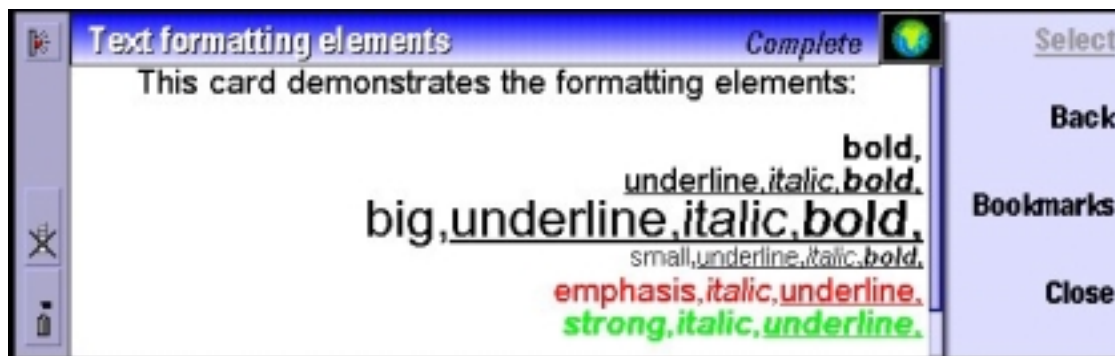#### 4.2.1 Text formatting

Figure 4-2   Card demonstrating text formatting elements

The browser supports all text formatting elements of WML. Emphasis and strong elements are shown in colours. The picture above demonstrates how different formatting elements are rendered as text, also specified in the table above.

Example 4-1   Text formatting elements

```
<wml>
<card id="text" title="Text formatting elements">

<p align="center">
  This card demonstrates the formatting elements:<br/>
</p>

<p align="right">
  <b>bold,</b><br/>

  <u>underline,<i>italic,<b>bold,</b></i></u><br/>

  <big>big,<u>underline,<i>italic,<b>bold,</b></i></u>
  </big><br/>
```

```
    <small>small,<u>underline,<i>italic,<b>bold,</b></i>
    </u></small><br/>

    <em>emphasis,<i>italic,</i><u>underline,</u></em>
    <br/>

    <strong>strong,<I>italic,</i><u>underline,</u>
    </strong><br/>
</p>

<do type="prev">
          <prev/>
</do>

</card>
</wml>
```

4.2.2 Paragraph

Paragraph <p> determines the text groups. A new paragraph always starts on a new line. Text inside a paragraph can be aligned to the left, center, or right, but left alignment is used by default. Alignment is determined in paragraph attributes (for example, <p align="right">).

4.2.3 Line Break

New lines in text can be defined by the line break element <br>. The <br> element can be used within other elements as long as it belongs to a paragraph.

4.2.4 Fieldset

The <fieldset> -element can be used for grouping elements. It implies a paragraph break between the elements, and the title of the element is used as the active element's title if it has no title of its own. Note that fieldset title is not supported for text, that is, text in a card can not be given a title by using fieldset element.

4.2.5 Text wrapping mode

4.3 Tables

Figure 4-3   A sample table.

A table element `<table>`, and its content elements `<tr>` and `<td>` are used to create columns and rows of text and images in a card surrounded by a gray border. Table elements do not specify column or intercolumn widths; the width of the column is same as the width of the widest cell in the column. If there is a title attribute on the table element, it will be rendered as the first row of the table, with a bold font and white font colour on a blue background. There can be multiple tables in one card. A line break is inserted before and after each table. The table may also contain images and links, which are focused from left to right and from top to bottom order. The maximum number of columns in a table is 30. All cell contents of tables over 30 columns are rendered in the last column of the table.

Example 4-2   `<table>` element

```
           <wml>
<card id="card1" title="Table element">
<p>

<table columns="4" align="LCCL" title=Weather Forecast>
<tr>
  <td><b>Date</b></td><td><b>F&apos;cast</b></td><td><b>T&#xB0;C
  </b></td><td><b>Description</b></td>
</tr>
<tr>
  <td>M 6/7</td><td><img src="rainy.gif" alt="rain" width="30"/>
  </td><td>Hi 20&#xB0;C, Lo 3&#xB0;C</td><td>Rainy</td>
</tr>
<tr>
  <td>T 6/8</td><td><img src="partcldy.gif" alt="part cldy"
  width="30"/></td><td>Hi 25&#xB0;C, Lo 15&#xB0;C</td>
  <td>Partly cloudy</td>
</tr>
<tr>
  <td>W 6/9</td><td><img src="cloudy.gif" alt="cloudy" width="30"
  /></td><td>Hi 22&#xB0;C, Lo &#xB0;C</td><td>Cloudy</td>
</tr>
<tr>
  <td>T 6/10</td><td><img src="rainy.gif" alt="rainy" width="30"
  /></td><td>Hi 20&#xB0;C, Lo 3&#xB0;C</td><td>Rainy</td>
</tr>
<tr>
```

```
  <td>F 6/11</td><td><img src="sunny.gif" alt="sunny" width="30"
  /></td><td>Hi 28&#xB0;C, Lo 0&#xB0;C</td><td>Sunny</td>
</tr>
</table>

</p>

<do type="prev">
        <prev/>
</do>

</card>
</wml>
```

4.4 Images, links, and timers

4.4.1 Img – Images

Figure 4-4   Different types of images

In addition to WBMP image format Nokia 9210 Communicator supports all the most commonly know image formats. Note: some WAP Gateways may not support all these content types. Nokia WAP Gateway products support all these content types listed here:

- a WBMP (Wireless Bitmap), "image/vnd.wap.wbmp"
- a JPEG File (incl. progressive), ".JPG", "image/jpeg"
- a GIF file, ".GIF", "image/gif"
- a PNG file, ".PNG", "image/png"

The image element is used for displaying an image in a card. Image sizes can be 60 kbytes or less. Although there are no absolute limits to an image's physical measurements. However, for taller images, the user must scroll down and for images wider than the browser area, images are scaled down to fit the screen. Images can also be used inside a table (see example 4-2) and as links (see example 4-4).

The img element should have width and height attributes defined in pixels. If either one or both of the width and height attributes have been defined in pixels, they are used as the size of the image and the placeholder. A fetched image is scaled to fit the specified size.

Width and height attribute values specified in percentages are ignored. An image is always forced into the horizontal space available. That is, if the image is wider than the screen, it will be scaled (width and height uniformly) to fit the horizontal space available.

If the horizontal margin (hspace) or vertical margin (vspace) has been defined for the img element, the number of pixels specified are left as a margin between the surrounding text flow and the image. Again, hspace and vspace specified in percentages are ignored.

When the user focuses on an image and it has not been (completely) fetched and is not inside an anchor element, the first CBA button will change to "Load Image". If the user selects it, the image will be fetched and displayed as it arrives from the network (or cache). If the image is inside an anchor or A element, the anchor's UI behaviour will override the image's behaviour.

If the user focuses on a completely loaded image, or if the focus is on an image that is being fetched and the fetch is completed, the first CBA button will change to "Save Image". If the user presses it, the standard Save As dialog box opens and the user can save the image in its original format (WBMP).

Example 4-3   Image element

```
<wml>
<card id="card1" title="Image element">

<p>
  Format:GIF <img src="games.gif" alt="Games"/>
  JPEG<img src="ClubN.jpg" alt="ClubNokia"/>
  <br/>
  Stretched JPEG:<img src="N6210.jpg" alt="Nokia 6210 - Streched!"
  width="150"/>
</p>

<do type="prev">
  <prev/>
</do>

</card>
</wml>
```

4.4.2 Anchors

A and anchor elements specify the head of links to another card or deck, for example. A link provides easy navigation through an application. An anchor may consist of text, an image, or a combination of text and an image. The <a> element is a short form of the anchor element. In general it is recommended to use the <a> element instead of anchor where possible. An unselected anchor is rendered as underlined (an image is framed), while the colour of a selected anchor is inverted.

Figure 4-5   Text and image links. An image within a table selected.
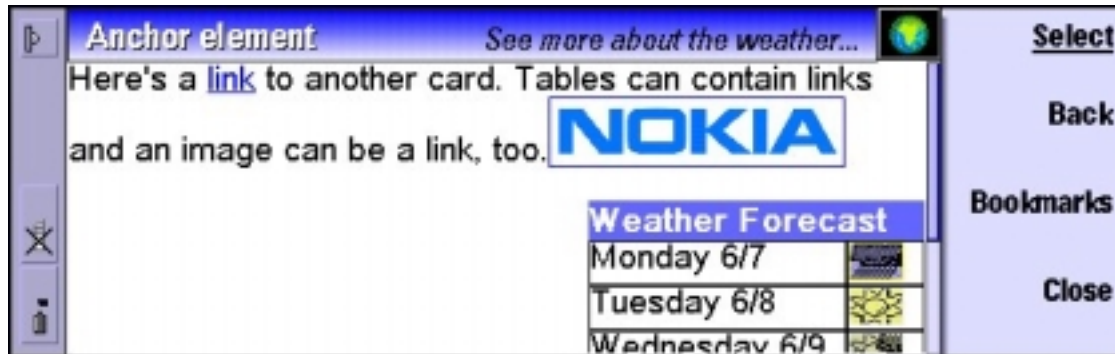
Example 4-4   Anchor element

```
            <wml>

<template>
  <do type="prev" label="go">
          <prev/>
  </do>
</template>
<wml>
<card id="card1" title="Anchor element">

<p>
Here's a 
<a title="Link to Products">link<go href="#products"/></a>
 to another card. Tables can contain links and an image can
be a link, too.
<a title="Link to Nokia Group" href="#group"><img src=
"Nokia_n1.gif" alt="nokia"/></a>
</p>

<p align="right">
<table columns="2" title="Weather Forecast">
  <tr>
  <td>Monday 6/7</td>
  <td><a title="See more about the weather..." href="#rain">
  <img src="rainy.gif" alt="rainy" width="30"/></a></td>
  </tr>

  <tr>
  <td>Tuesday 6/8</td>
  <td><img src="sunny.gif" alt="sunny" width="30"/></td>
  </tr>

  <tr>
  <td>Wednesday 6/9</td><td><img src="partcldy.gif" alt="part
  cldy" width="30"/></td>
  </tr>
```

```
   <tr>
   <td>Thursday 6/10</td>
   <td><img src="cloudy.gif" alt="cloudy" width="30"/></td>
   </tr>

   <tr>
   <td>Friday 6/11</td><td><img src="rainy.gif" alt="rainy"
   width="30"/></td>
   </tr>
</table>
</p>

</card>
</wml>
```

4.4.3 Timer

The timer element defines a card timer, which processes inactivity or idle time. Timer starts the count immediately after the card is loaded. This element can be used only once in card and its unit is 1/10 s.

4.5 Dynamic buttons

4.5.1 Do elements

The do element binds a task to a user action, which can be executed in an options list on the left side of the screen. The options list contains all currently accessible do elements (options). The text representing each do element is the do element's label attribute value, or if it has not been defined, the UI string corresponding to the element's type attribute value. The labels are updated with current values whenever a refresh task is executed but the focus in a list is not changed.
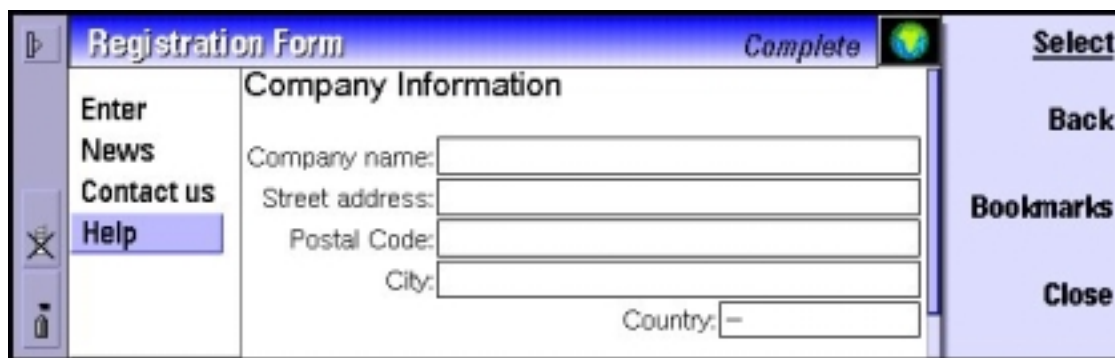

Figure 4-6   Do elements in an options list. The options list is focused.

Each WML deck can have a template element. The template contains do elements that are to be displayed on each card of the deck. A card's do elements can either override or hide the template's do elements. Hidden do elements will not be shown.

The order of the card's do elements in the options list is the same as in which they are defined on the card, unless they override the template's do elements. The template's do elements are last on the options list, that is, after all the card's do elements. If a do element on a card overrides the do element on a template, the card's do element replaces the template's do element in the list.

All optional do elements are displayed. Do elements of the type "prev" are handled as an exception. (See section 4.5.2)

Example 4-5   Do element - options list items

```
<wml>
<template>
<do type="accept" name="news" label="News">
<go href="http://.../news.wml"/></do>
<do type="accept" name="contactus" label="Contact us">
<go href="http://.../contact.wml"/></do>
<do type="accept" label="Help" name="helphelp" optional="false">
<go href="http://.../help.wml"/></do>
</template>

<card id="form" title="Registration Form">
<p>
Company Information<br/>
</p>

<p align="right">
<small>Company name:</small>
<input name="company" type="text" title="Company name" format="*A"
  emptyok="false" size="25"/><br/>
<small>Street address:</small>
<input name="address" type="text" title="Street address"
  emptyok="false" size="25"/><br/>
<small>Postal Code:</small>
<input name="code" type="text" title="Postal code" emptyok="false"
  size="25"/><br/>
<small>City:</small>
<input name="city" type="text" title="City" emptyok="false"
  size="25"/><br/>
<small>Country:</small>

<select name="country" multiple="true">
<option value="-">-</option>
<option value="Australia">Australia</option>
<option value="Canada" size="25">Canada</option>
<option value="Denmark">Denmark</option>
<option value="Finland">Finland</option>
<option value="France">France</option>
<option value="Germany">Germany</option>
<option value="Italy">Italy</option>
</select>
</p>

<do type="accept" label="Enter" optional="false">
<go href="#confirm" method="get" sendreferer="false"/>
</do>

<do type="prev" label="Back" optional="false">
<prev/>
</do>
```

```
</card>
</wml>
```

4.5.2 Prev type do elements

Whenever a new card is shown, a new history list item is created. When activating the prev task on a WML card, the target URL is the previous history list item and the current card (in which prev was activated) is not retained in the history list (there is no "forward" function).

Do elements of the type "prev" are handled as an exception to other types of do elements (see section 4.5.1). If there is only one "prev" type do element active, it is not shown on the options list. Instead, it is bound to the 2nd CBA button and it is labelled as "Back". The 2nd CBA is dimmed if there is zero or more than one "prev" type do element active and do elements are displayed on the options list like all other do elements. It is likely that most pages implement only one do element of the type "prev".

The author of the WAP service is always responsible for the working navigation model within the service. If there is a need for back functionality, it should be explicitly defined with the "prev" type do element. The WAP browser does not have any implicit back functionality!

4.6 WML Input processing

4.6.1 Input elements

Figure 4-7   Input element – focus is in the input box with numeric format.

An input element specifies a text entry object for data input. It is shown as an input box in a card. The text entered in the input element can be determined by the format attribute. The format string specifies that a particular character must be a number, an uppercase character, or a lowercase character, for example.

The user can enter any characters in the input field but the characters that do not currently match the format are drawn in red colour and are underlined (so that incorrect spaces or missing characters become visible. See section 4-7 above). In addition, every time the user enters a different type of input than the browser was expecting, an appropriate info message in an "Incorrect input dialog box" will be displayed. All characters that do match the format are drawn in black (normal body text colour).

The author can specify a maximum length for the entry either with the format or with the maxlength attribute. If the maximum length is reached, the input element will not insert any characters in the field, but will beep and display an info message.

The author can specify with the emptyok attribute that the entry must match the format mask. If there are input elements that do have that requirement, but the entry in those elements does not match the mask (such as when the value is empty) and a task is initiated (either by a user action or timer), the Missing Input dialog box will appear. Once the user has closed the dialog box, the focus will move to the first of those input elements on the card. Note that the user can exit the card only if all input elements have a valid value or if they are allowed to be empty.

Example 4-6   Input element – Input box

```wml
<wml>
<card id="Form" title="Registration Form">

<p>
Please login:<br/>
</p>

<p align="right">
  <small>User name:</small>
  <input name="user" type="text" title="Max. 8 char. long"
    format="*A" emptyok="false" maxlength="8" size="25"/><br/>
  <small>User password:</small>
  <input name="user pw" type="password" title="Max 16 char. Long"
    format="*A" emptyok="false" maxlength="16" size="25"/><br/>
</p>

<p>
  Input your birthday:<br/>
</p>

<p align="right">
  <small>Date of birth:</small>
  <input name="birth" type="text" title="dd/mm/yyyy" format=
    "NN\/NN\/4N" emptyok="false" maxlength="14 size="25"/><br/>
</p>

<do type="accept" label="Enter">
  <go href="#confirm" method="get"/>
</do>

<do type="prev" label="Back">
  <prev/>
</do>

</card>
</wml>
```
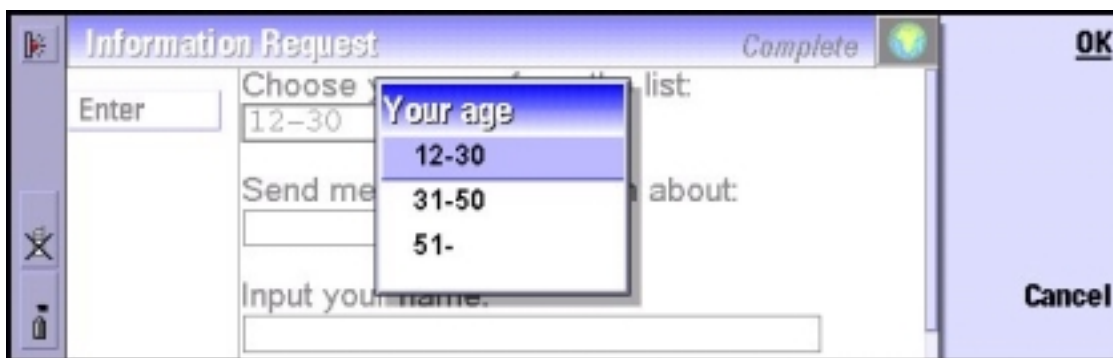
4.6.2 Selection lists

Figure 4-8   Simple single-selection list

Selection lists are an input element that specifies a list of options for the user to choose from. The selection list can be either single-selection or multiple-selection element. Each option is specified by an option element. Options can be organised into groups by using the optgroup element.

A selection list is opened by focusing on the input box and pressing the 1st CBA button for "Change". The select list appears. A single-selection list has a title of the title attribute, and if it hasn't been defined, the title displays the text "Selection" by default. A multiple-selection list has two windows; the left one is for unselected items and the right one contains the selected items. The title of the non-selected items pane is the select element's title or if one has not been defined, the default text displayed is "Not selected". The title of the selected items pane is "selected". The user can switch focus between the two dialog boxes with the tab key.

If the options have been grouped with an optgroup element, the optgroup title will be shown as a bold item in the selection list. All options under the optgroup element are intended and shown under the optgroup item. Note that the Nokia 9210 Communicator only supports one level of optgroups. Optgroup on a deeper level is ignored and its options are handled as if they were under the first level optgroup (that is, intended as one level).

Figure 4-9   Multiple-selection list

Example 4-7   Selection lists

```
<wml>
<card id="phones" title="Information Request">

<p align="left">
```
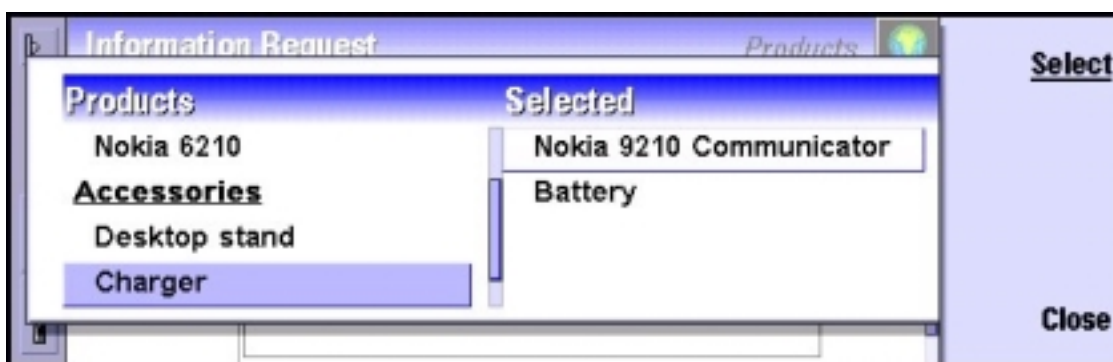
```
Choose your age from the list:<br/>
<select title="Your age">
<option title="12-30">12-30</option>
<option title="31-50">31-50</option>
<option title="51-">51-</option>
</select><br/>

<br/>Send me more information about:<br/>
<select name="models" title="Products" multiple="true">

<optgroup title="Phones">
<option>Nokia 3310</option>
<option>Nokia 6210</option>
<option>Nokia 9210 Communicator</option>
</optgroup>

<optgroup title="Accessories">
<option>Battery</option>
<option>Desktop stand</option>
<option>Charger</option>
<option value="Headset">Headset</option>
</optgroup>

</select><br/>

<br/>Input your name:<br/>
<input name="name" type="text" title="Name" size="30"/>
</p>

<do type="accept" label="Enter" optional="false">
  <go href="#confirm" method="get" sendreferer="false"/>
</do>

<do type="prev" label="Back" optional="false">
  <prev/>
</do>

</card>
</wml>
```

4.7 Titles and Labels

4.7.1 Titles

A title bar is divided in two parts, on the left is the card title and on the right is space for element titles, which are shown only when the element is focused. The card title is displayed continuously and if there is no card title, the actual URL of the card is displayed instead. Below is a list of element titles, which use the right side of the title bar.

- Link, attribute "title"
- Image, attribute "alt"
- Input, attribute "title"

- Select, attribute "title"

In addition, there are elements, which have a title, but it is not shown on the main title bar. These are tables whose title attribute value is shown on its own title bar, select elements title attribute value is shown on main title bar when the input box is focused. After opening the select list, the same title is shown on the select lists own title bar. The optgroup title attributes value is shown as a bold item in select list.

4.7.2 Labels

The only element with a label attribute is the do element. Cards currently accessible do elements are listed in the options list on the left side of application area, where their label attribute values are shown. An exception to this is the "prev" type do element (see section 4.5.2).

4.8 Limitations

This section describes technical limitations of the WAP services application.

The limitation for encoded WML file is 64 kbytes.

4.8.1 WML character sets.

The WAP browser uses the character sets defined and used in the EPOC platform, this includes the usage of UNICODE as necessary. Where multiple character sets are found the reference-processing model specified in Section 6.1 of the WML specification is used.

4.8.2 Features not supported in WML 1.1

'class' attributes as described in the WML Specification, section 8.9.

5. GENERAL USABILITY ISSUES – GUIDELINES FOR DESIGN

WML History Navigation

Due to the differences in the processing of the history between WAP and WEB, only the history processing described in the WML specifications is followed. This means that the service provider has to take care of the navigation as a whole!

Nokia Mobile Phones is renowned for their simple to use and intuitive user interface. In order to create a service that will be perceived as usable and easy to understand, Nokia Mobile Phones provides some general guidelines to help service providers develop their applications.

5.1 Validate Your WML

There are several XML validators available that validate your documents against WML Document Type Definition. It is recommended that authors validate their WAP pages, because invalid WML is always treated as an error and discarded (that is, it is not shown to the user!).

Several XML validators can be found at http://www.wapdevelopers.org/xml.tpl?CALLER=index.tpl.

5.2 Site Organization

5.2.1 Avoid 'doormat' pages

The user accesses your WAP site over a GSM data call, and pays per second. It is not recommended that your site begins with a 'doormat' page, which serves no purpose other than perhaps to greet the visitor and to display a logo. It is better that your site allows users to access your service directly. If there is a need to use a timer, it should not take longer than ~1.5 seconds, if it takes a longer time, add a 'continue' link to give the user possibility to exit the card.

5.2.2 Automatically Detect the Browser

If you are using a modern Web site hosting environment, it is possible to detect the browser as well as the language mode and supply the correct content transparently, without requiring user interaction. The Nokia 9210 Communicator display is larger than normal phone displays, and thus has more possibilities that are worth taking advantage of.

5.2.3 Optimise for Size

The size of the content is critical. If you have large decks (listings, large tables, etc.), consider splitting them in multiple parts for faster downloading.

As for the total downloading time, some studies place an upper limit for acceptable delay to 10 to 15 seconds, including all images, on a PC-based browser. It is highly recommended to have a response time (excluding connection time) of less than 10 seconds in mobile environments.

5.2.4 Choose Descriptive Card Titles

It is highly useful to give a descriptive name for the card. It might be a good idea to start the title with your service's name and keep the total length of the title short.

It also pays to use meaningful URLs since the user sees the URL of the currently selected link on the screen and can use it as a navigational help, especially when images have not been loaded.

5.2.5 Pay Attention to the First Screenful

Because of the communicator's form factor, the first (topmost) screenful of any page is the most important one. All of the often-used navigational links, search fields, login screens, and bulk of the information should reside there, if at all. The user is then able to navigate forward before the rest of the card has been loaded, and the user does not have to scroll the card.

5.2.6 Use of do elements

Always use descriptive and short labels for all do elements. Favor local do elements instead of anchored links that are totally out of context. Always include the prev element in each card to enable backward navigation.

5.3 Pictures, Tables, and Colour

5.3.1 Avoid Useless Images

Downloading images takes time, and many users may switch off the loading of images for more speed. Try to optimise the size of images. If you have large images on your site, consider using thumbnails for the image index.

Always provide an alternative text (using the ALT attribute of the IMG element) for images that convey information. Always use a null alternative text (ALT="") for images which do not convey information, or are used for page layout or decorative purposes only.

5.3.2 Use Reasonable Image Sizes

The maximum image width on the WAP browser screen is 490 pixels. Images larger than that will be automatically scaled down to fit the screen. Using any larger images than 490 pixels is a waste of bandwidth and will slow down the service connection. A maximum size for images downloaded through data calls is about 60kb although 30kb is recommended maximum.

5.3.3 colormap

WAP does not include any support for colours, but the Nokia 9210 Communicator's WAP browser supports image formats with colour (other than the WBMP image format).

The Nokia 9210 Communicator supports 4096 colours. The true colour is specified by 24 bits, 8 bits for each colour component (red, green, and blue). The Nokia 9210 Communicator uses the most significant 4 bits of each part of the colour component, resulting in a colour depth of 12 bits. All other colours will either be dithered or mapped to the closest colour available, depending on the application.

When creating graphics with uniform fills, use only the colours, which will not be dithered. Some image processing tools may be able to reduce the size of images if you restrict the number of colours to the exact 4096 colours supported by the Nokia 9210 Communicator.

Do not use colours, which are too close to each other (colours that differ only in the least significant four bits of each colour component). This may result in the colours being mapped to the exact same colour.

5.3.4 Use Reasonable Table Sizes

If the table size exceeds the maximum width of the WAP services application screen due to the number of columns, the horizontal scroll bar will appear. Anyhow, to keep the cell content readable, special attention should be paid on table structuring.

5.4 Use of card titles and element labels

Card titles describe the content of the display and their use is recommended. They help the user to navigate because they function as a reminder of where the user is in the application. The header text should be determined by the item previously selected by the user. For instance, the card title 'Bookmarks' tells the user that the display contains a list of bookmarks in the application and that the options item previously selected was *Bookmarks*.

Proportional fonts are used in header texts, and if the header text is too long, it is automatically truncated. Truncation is usually better than abbreviation, because the user might be confused by unfamiliar abbreviations that are difficult to understand.

5.5 Perform usability test

It is always good to perform a usability test for new applications. People who have not been involved in the design or development of the application tend to notice potential usability problems often not obvious to those who know the design by heart. Usability tests should always be performed as early as possible in the development process. Any necessary changes resulting from the tests can then be implemented within the development schedule. Try to recruit users who are representative of the end users of the application, and try to conduct the usability test on a smaller scale, if the schedule does not allow for extensive testing.