

## **Deviations and Limitations in WapIDE 2.1**

### **Abstract**

This document lists a few areas of functionality in which the WapIDE 2.1 device simulation differs from the Ericsson R320s WAP enabled phone.

Please remember that when this document is written the R320 is still under development and the characteristics can change somewhat when released.

### **Contents**

1	WML Interpretation and rendering .....	2
1.1	Optgroup .....	2
1.2	Option .....	2
1.3	Fieldset .....	2
1.4	Nowrap text Mode .....	2
1.5	Input element values .....	2
1.6	Soft-hyphen and Non-breaking space .....	3
1.7	Images .....	3
1.8	Tables .....	3
1.9	Control Elements.....	3
1.10	Markup .....	4
2	Document encoding .....	4
3	HTTP/WSP Headers .....	4
3.1	Accept-Language .....	4
4	Tracing.....	4
5	Cacheing.....	5
6	Miscellaneous Issues .....	5
6.1	Content Types and other Limitations .....	5
6.2	Maxfloat .....	5
6.3	Wmlib .....	5
6.4	Application Designer .....	5
7	References.....	5

## 1 WML Interpretation and rendering

### 1.1 Optgroup

The **<optgroup>** tag is ignored when rendering options in select elements. This means that any structure defined by grouping is lost and the options flattened and presented on the same level. Any title attribute specified in the **<optgroup>** tag is thus not presented.

The R320s handles this tag in a similar manner.

### 1.2 Option

WapIDE uses the **title** attribute or a generic label if needed in option elements. This is a deviation from the real R320 which does not.

### 1.3 Fieldset

The **<fieldset>** tag is ignored when rendering elements that have been explicitly grouped. The small screen area does not allow a meaningful implementation of this grouping. Any title attribute specified in the **<fieldset>** tag is thus not presented.

The R320s handles this tag in a similar manner.

### 1.4 Nowrap text Mode

The use of **mode="nowrap"** in **<p>** tags on text lines that are longer than the width of the screen, will, as in the R320s, scroll the text horizontally when displayed on the second line of the screen.

The implementation of this mechanism in WapIDE, may nevertheless differ slightly from that of the R320s.

### 1.5 Input element values

Entry of text or numerical information is always performed in a pop-up window in which the characters are echoed as they are entered. When the input is entered the prompt and the entered value are presented on a single line.

In WapIDE the following algorithm is used when these two items do not fit on a single line of text:

1. If necessary the prompt text is truncated to leave space for at least 3 characters of the value.
2. The value is presented within square brackets and truncated to fit on the same line as the prompt text. At least 3 characters from the value will be shown.

The R320s phone does not truncate the prompt text but truncates the value so that it will fit on the same line as the prompt text.

## 1.6 Soft-hyphen and Non-breaking space

The soft-hyphen character entity (**&shy;** or **&#173**) and the non-breaking space (**&nbsp;** or **&#160**) entities do not work properly.

The R320s device does interpret these entities as is intended by the specification, see ref. [1].

## 1.7 Images

The **align**, **height** and **width** attributes of the **<image>** tag are not fully implemented. Alignment is always **center** regardless of what is specified by the **align** attribute.

The height is always allocated as a multiple of the font height based on the image bitmap height regardless of the **height** attribute. The reason for this is because the image is scrolled one text line at a time and because text and images cannot be placed on the same line.

The width is calculated from the width of the image bitmap and if it exceeds the width of the display area is truncated to 90% of this area regardless of the **width** attribute.

The same limitations are valid for the R320s device.

The **localsrc** attribute in images is ignored.

## 1.8 Tables

The rendering of tables in WapIDE 2.0 is very similar to that of the R320s with scrolling columnwise horizontally.

Vertical scrolling differs since in the R320s vertical scrolling is always performed one text line at a time, i.e 13 pixels. In WapIDE this is performed one table row at a time which may be more than one line of text.

Navigation in tables that contain links (a tags) is in WapIDE a bit peculiar due to the dual purpose of the up/down keys. These keys are used for scrolling the table vertically **and** for moving to the previous/next link.

The conflict is settled in favour of link navigation. The up/down keys will move to the next link if there is one unless it's on another table row. If it is, the up/down key will show column 1 of the previous/next table row instead.

This makes navigation in large tables with links a little easier but it may differ from the behaviour of the real R320s terminal.

## 1.9 Control Elements

The **tabindex** attribute, used in the input elements (e.g. **select** and **input**) is ignored, this is an optional attribute and is also ignored in the real device R320.

Tabbing is a navigational accelerator and is optional for all user agents.

## 1.10 Markup

The **big** and **small** markup attributes are in WapIDE 2.0 as well as the R320s terminal ignored since only one font size is supported. The **<u>** tag is ignored in the R320s phone but implemented in WapIDE. The **<em>** tag is also interpreted slightly differently compared to the real R320s device.

The **xml:lang** attribute is ignored.

## 2 Document encoding

The **charset** parameter placed in **meta** information inside the document is ignored which is a deviation from chapter 6.1 in ref. [1]. It is, however, supported in the normal way in the HTTP header and in the XML header declaration.

The encoding for WML and WMLScript source documents is assumed to be ISO-8859-1 (Latin 1) unless this is explicitly specified by HTTP headers or in the WML deck itself in the XML header declaration.

The encoding WML and WMLScript bytecode documents is assumed to be UTF-8 unless explicitly specified by the HTTP/WSP headers.

## 3 HTTP/WSP Headers

### 3.1 Accept-Language

The **Accept-Language** header is not used in requests.

## 4 Tracing

If the WML Variable Viewer or log is turned on and a variable contains a Tcl sensitive character such as “ ‘ { } , WapIDE might crash. The problem only occurs when the log or viewer is activated.

## 5 Cacheing

Responses to normal HTTP GET requests that are returned with status codes 200 are kept in a cache area. Exceptions to this are:

- Responses from POST requests
- Responses from GET requests with arguments
- Responses with the cache-control directive **no-cache** specified in **meta** data or in HTTP headers.
- Responses with status codes 201-206

## 6 Miscellaneous Issues

### 6.1 Content Types and other Limitations

Below are listed some unsupported functionality and limitations that apply to WapIDE 2.1:

- The Content-types **vcal** and **vcard** are not supported.
- There is no support for the PNG image encoding (**image/png**).
- WapIDE uses HTTP/1.0 instead of HTTP/1.1

### 6.2 Maxfloat

The **maxfloat** symbol used in WMLScript functions, is internally represented with only 6 digit accuracy rather than with 9 digits as expected. As a result comparisons with 9 digit literals will incorrectly fail.

### 6.3 Wmlib

The **wmlib** directory in **servtools** contains a Perl 5.0 function library with functions for generating WML 1.1 content. There are, however, not yet any functions in this library that support tables.

### 6.4 Application Designer

The drop-down file selection widget in the Application Designer editor does not work properly in the Windows version.

## 7 References

- [1] Wireless Markup Language Specification 1.1, 1999-06-16,  
[http://www.wapforum.org/what/technical\\_1\\_1.htm](http://www.wapforum.org/what/technical_1_1.htm)