© 2009 Research In Motion Limited. All rights reserved. BlackBerry®, RIM®, Research In Motion, SureType®, SurePress™ and related trademarks, names and logos are the property of Research In Motion Limited and are registered and/or used in the U.S. and countries around the world.

Apache Tomcat is a trademark of The Apache Software Foundation; Bluetooth is a trademark of Bluetooth SIG; GSM and Global System for Mobile communications are trademarks of the GSM MOU Association; IBM, Domino, Lotus, Lotus

s, Lotus Organizer and Sametime are trademarks of International Business Machines Corporation; IEEE is a trademark of the Institute of Electrical and Electronics Engineers, Inc.; Intellisync is a trademark of Intellisync Corporation; Java, J2ME, JavaScript, JDBC, and JRE are trademarks of Sun Microsystems, Inc.; Microsoft, Active Directory, Active X, Entourage, Excel, Internet Explorer, Outlook, PowerPoint, SQL Server, Visual Studio, Windows, Windows Media, Windows Mobile and Windows Server are trademarks of Microsoft Corporation; Novell and GroupWise are trademarks of Novell, Inc.; PGP is a trademark of PGP Corporation; Wi-Fi is a trademark of the Wi-Fi Alliance; All other trademarks are the property of their respective owners.

The BlackBerry smartphone and other devices and/or associated software are protected by copyright, international treaties, and various patents, including one or more of the following U.S. patents: 6,278,442; 6,271,605; 6,219,694; 6,075,470; 6,073,318; D445,428; D433,460; D416,256. Other patents are registered or pending in the U.S. and in various countries around the world. Visit www.rim.com/patents for a list of RIM (as hereinafter defined) patents.

This documentation including all documentation incorporated by reference herein such as those provided or made available by hyperlink is provided or made accessible "AS IS" and "AS AVAILABLE" and without condition, endorsement, guarantee, representation or warranty of any kind and/or liability, be it express or implied, including, but not limited to, liability for breach of contract, negligence, absolute, special, incidental, or consequential damages, or damages for lost profits or revenues, failure to realize any expected savings, business interruption, loss of business opportunity, or corruption or loss of data, failures to transmit or receive any data, problems associated with any applications used in conjunction with RIM Products or Services, downtime costs, loss of the use of RIM Products or Services or any portion thereof or of any airtime services, cost of substitute goods, costs of cover, facilities or services, cost of capital, or other similar pecuniary losses, whether or not such damages were foreseen or unforeseen, and even if RIM has been advised of the possibility of such damages.

TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW IN YOUR JURISDICTION, RIM SHALL HAVE NO OTHER OBLIGATION, DUTY, OR LIABILITY WHATSOEVER IN CONTRACT, TORT, OR OTHERWISE TO YOU INCLUDING ANY LIABILITY FOR NEGLIGENCE OR STRICT LIABILITY.

THE LIMITATIONS, EXCLUSIONS, AND DISCLAIMERS HEREIN SHALL APPLY: (A) IRRESPECTIVE OF THE NATURE OF THE CAUSE OF ACTION, DEMAND, OR ACTION BY YOU INCLUDING BUT NOT LIMITED TO BREACH OF CONTRACT, NEGLIGENCE, TORT, STRICT LIABILITY OR ANY OTHER LEGAL THEORY AND SHALL SURVIVE A FUNDAMENTAL BREACH OR BREACHES OR THE FAILURE OF THE ESSENTIAL PURPOSE OF THIS AGREEMENT OR ANY REMEDY CONTAINED HEREIN; AND (B) TO RIM AND ITS AFFILIATED COMPANIES, THEIR SUCCESSORS, ASSIGNS, AGENTS, SUPPLIERS (INCLUDING AIRTIME SERVICE PROVIDERS), AUTHORIZED RIM DISTRIBUTORS (ALSO INCLUDING AIRTIME SERVICE PROVIDERS) AND THEIR RESPECTIVE DIRECTORS, EMPLOYEES AND INDEPENDENT CONTRACTORS.
IN ADDITION TO THE LIMITATIONS AND EXCLUSIONS SET OUT ABOVE, IN NO EVENT SHALL ANY DIRECTOR, EMPLOYEE, AGENT, DISTRIBUTOR, SUPPLIER, INDEPENDENT CONTRACTOR OF RIM OR ANY AFFILIATES OF RIM HAVE ANY LIABILITY ARISING FROM OR RELATED TO THE DOCUMENTATION.

Prior to subscribing for, installing or using any Third Party Products and Services it is your responsibility to ensure that your airtime service provider has agreed to support all of their features. Some airtime service providers may not offer Internet browsing functionality with a subscription to BlackBerry® Internet Service. Check with your service provider for availability, roaming arrangements, service plans and features. Installation or use of Third Party Products and Services with RIM's products and services may require one or more patent, trademark, copyright or other licenses in order to avoid infringement or violation of third party rights. You are solely responsible for determining whether to use, Third Party Products and Services and if any third party licenses are required to do so. If required you are responsible for acquiring them. You should not install or use Third Party Products and Services until all necessary licenses have been acquired. Any Third Party Products and Services that are provided with RIM's products and services are provided as a convenience to you and are provided "AS IS" with no express or implied conditions, endorsements, guarantees, representations or warranties of any kind by RIM and RIM assumes no liability whatsoever, in relation thereto. Your use of Third Party Products and Services shall be governed by and subject to you agreeing to the terms of separate licenses and other agreements applicable thereto with third parties, except to the extent expressly covered by a license or other agreement with RIM.

Certain features outlined in this documentation require a minimum version of BlackBerry® Enterprise Server, BlackBerry® Desktop Software, BlackBerry® Device Software and/or BlackBerry Handheld Software and may require additional development or Third Party Products and Services for access to corporate applications.

Certain products mentioned in this documentation include software developed by the Apache Software Foundation (http://www.apache.org/) and/or licensed pursuant to Apache License, Version 2.0 (http://www.apache.org/licenses/). For more information, see the NOTICE.txt file included with the software. Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

The terms of use of any RIM product or service are set out in a separate license or other agreement with RIM applicable thereto. NOTHING IN THIS DOCUMENTATION IS INTENDED TO SUPERSEDE ANY EXPRESS WRITTEN AGREEMENTS OR WARRANTIES PROVIDED BY RIM FOR PORTIONS OF ANY RIM PRODUCT OR SERVICE OTHER THAN THIS DOCUMENTATION.
Contributors

Many people were involved in the development of this book, and we want to extend our thanks to them—our apologies if we have forgotten someone!

Certification Team
Kristy Grieve

Content Development Team
Mary Pat Hinton
Jamie Jewer
Wayne Annable (Innovatia)
Angela-Jo Griffin (Innovatia)
Ellen Lupu (Innovatia)
Steve Malone (Innovatia)
Laura Mercer (Innovatia)
Erin Peterson (Innovatia)
David Watson (Innovatia)
Brian White (Innovatia)

Graphic Design
Elizabeth Zwiers
Greg Young

Legal Review
Shalini Maharaj-Dookeran

Technical Review (Subject Matter Experts)
Stan Devitt
Mark Sohm
Jennifer Emery
Adam Abramski
Rob Kline
Mark Rathwell
Michael Clewley
John Svazic
Orlin Stoev
Tomasz Buczek
Nikos Kyprianou
Eric Chan
Raymond Mendoza
Mike Kirkup
Chris Bender
Mario Pereira-Bernardino
Tao Baecklund
Earl Oliver (University of Waterloo)
Michael Weitzel
Garett Beukeboom
Kevin Bater

**Project Business Coordination**
Christina Sookram
Tracey Weiler
Karen Moser
Suzanne Graham (Innovatia)
Content

Chapter 1: Introduction to Web based application development for mobile devices

Application design considerations for mobile devices................................................................. 8
Create custom application control policies for a BlackBerry Java Application .................................. 9
Designing web page content for the BlackBerry Browser ............................................................... 10
Designing site navigation for the BlackBerry Browser ...................................................................... 14
Adding form controls....................................................................................................................... 19
Creating and using images................................................................................................................ 23
BlackBerry Web Development Plug-in for Eclipse 1.0 ................................................................ 28
New Web Development Enhancements ......................................................................................... 28
Summary ........................................................................................................................................ 30
Review Questions ........................................................................................................................... 31

Chapter 2: Understanding Java APIs for mobile devices

Java APIs for mobile devices ........................................................................................................... 34
What is an API? ................................................................................................................................ 34
Java ME ............................................................................................................................................ 34
Supported Java APIs ....................................................................................................................... 35
Supported Java API extensions ......................................................................................................... 37
Summary ........................................................................................................................................ 38
Review Questions ........................................................................................................................... 39

Chapter 3: Developing a user interface for mobile devices

Introducing the BlackBerry smartphone UI ........................................................................................ 42
Explaining best practices for successful mobile device UIs ................................................................ 43
Components of the BlackBerry smartphone UI ............................................................................... 45
....................................................................................................................................................... 46
Quiz ................................................................................................................................................ 46
Create a basic UI for a mobile application ......................................................................................... 48
Screens .......................................................................................................................................... 48
Quiz............................................................................................................................................... 55
Managers ....................................................................................................................................... 57
Fields............................................................................................................................................ 58
Quiz .............................................................................................................................................. 68
Drawing directly on the screen .......................................................................................................... 70
Graphics class................................................................................................................................. 70
Drawing a circle............................................................................................................................... 73
<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Methods for testing applications</td>
<td>167</td>
</tr>
<tr>
<td>Preverifying applications</td>
<td>166</td>
</tr>
<tr>
<td>Creating and starting threads</td>
<td>139</td>
</tr>
<tr>
<td>Suspending threads</td>
<td>139</td>
</tr>
<tr>
<td>Prioritizing threads</td>
<td>140</td>
</tr>
<tr>
<td>Dangers of multithreaded applications</td>
<td>141</td>
</tr>
<tr>
<td>Stopping threads</td>
<td>140</td>
</tr>
<tr>
<td>Summary</td>
<td>142</td>
</tr>
<tr>
<td>Quiz</td>
<td>143</td>
</tr>
<tr>
<td>Review Questions</td>
<td>143</td>
</tr>
</tbody>
</table>

**Chapter 6: Network communication for mobile devices**

<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Network transport routes</td>
<td>148</td>
</tr>
<tr>
<td>WAP network gateways</td>
<td>148</td>
</tr>
<tr>
<td>The BlackBerry MDS connection service network gateway</td>
<td>150</td>
</tr>
<tr>
<td>The BlackBerry Internet Service Browsing network gateway</td>
<td>151</td>
</tr>
<tr>
<td>Direct TCP networks</td>
<td>152</td>
</tr>
<tr>
<td>Wi-Fi connections</td>
<td>153</td>
</tr>
<tr>
<td>Least cost routing</td>
<td>155</td>
</tr>
<tr>
<td>Serial bypass</td>
<td>156</td>
</tr>
<tr>
<td>Creating an HTTP connection</td>
<td>157</td>
</tr>
<tr>
<td>Configuring a socket connection</td>
<td>159</td>
</tr>
<tr>
<td>Summary</td>
<td>162</td>
</tr>
<tr>
<td>Review Questions</td>
<td>163</td>
</tr>
</tbody>
</table>

**Chapter 7: Testing and debugging an application**

<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Preverifying applications</td>
<td>166</td>
</tr>
<tr>
<td>Methods for testing applications</td>
<td>167</td>
</tr>
<tr>
<td>Testing applications on a BlackBerry Smartphone Simulator</td>
<td>167</td>
</tr>
<tr>
<td>Testing applications on a BlackBerry smartphone</td>
<td>167</td>
</tr>
<tr>
<td>Quiz</td>
<td>170</td>
</tr>
<tr>
<td>Debugging tools available in BlackBerry JDE Plug-in for Eclipse</td>
<td>173</td>
</tr>
<tr>
<td>Console window</td>
<td>175</td>
</tr>
<tr>
<td>Variables and Debug window</td>
<td>175</td>
</tr>
<tr>
<td>BlackBerry Memory Statistics View</td>
<td>177</td>
</tr>
<tr>
<td>BlackBerry Objects View</td>
<td>179</td>
</tr>
<tr>
<td>BlackBerry Profiler View</td>
<td>181</td>
</tr>
<tr>
<td>Quiz</td>
<td>184</td>
</tr>
<tr>
<td>Garbage collection, memory leaks and deadlocks</td>
<td>187</td>
</tr>
<tr>
<td>Garbage collection</td>
<td>187</td>
</tr>
<tr>
<td>Memory leaks</td>
<td>188</td>
</tr>
<tr>
<td>Deadlocks</td>
<td>188</td>
</tr>
<tr>
<td>Quiz</td>
<td>190</td>
</tr>
<tr>
<td>Event Log</td>
<td>192</td>
</tr>
<tr>
<td>Summary</td>
<td>195</td>
</tr>
</tbody>
</table>
Preface

This textbook is one in a series offered by the BlackBerry® Academic Program. The BlackBerry Academic Program encompasses and facilitates the development, implementation, and assessment of academic programs for Research In Motion. The BlackBerry Academic Program supports students learning about BlackBerry solutions by developing opportunities to integrate educational components related to BlackBerry solutions within colleges and universities.

This textbook is designed to provide information to students about developing applications for BlackBerry smartphones and to help students prepare for the final course exam. A number of the topics in this textbook correspond to the topics found in the final exam, but this textbook also includes information on additional resources that can help students to prepare for the final exam.
Chapter 1

Introduction to Web based application development for mobile devices

Objectives

- Describe application design considerations for mobile devices
- Describe custom Java® applications
- Describe BlackBerry application development

This chapter outlines the design considerations and best practices for mobile BlackBerry® devices. It describes how to customize Java applications using Java development tools.
Application design considerations for mobile devices

Applications designed for BlackBerry devices should provide features that are simple to use, and preserve resources. When you design your BlackBerry device application, consider the differences between mobile devices and computers.

1. Features which consume a lot of resources are less likely to be welcome or usable on a mobile device.
2. The user’s goals are different.
3. The applications should be as consistent in design as possible.

Mobile devices have the following characteristics:
- smaller screen size that can display a limited number of characters
- slower processor speeds
- use wireless networks that have a longer latency period than standard LANs
- less available memory
- shorter battery life
- display one screen at a time

Mobile device users use applications on their mobile device differently than they use applications on a computer. On mobile devices, users expect to find information quickly. For example, a CRM system can provide a massive amount of information, but users only require a small amount of that information at one time. The BlackBerry UI is designed so that users can perform tasks easily and access information quickly.

When you design applications for BlackBerry devices, be as consistent as possible with other BlackBerry device applications.

Consider the following guidelines:
- Use or extend existing UI components where possible so that your application can inherit the default behavior of the component.
- Follow the standard navigation model as closely as possible so that users can make full use of the keyboard and trackball.
- Make all actions available from the menu. Verify that the actions available in the menu are relevant to the user’s current context.
- Support and extend user tasks in useful ways. For example, when users download an application, the application should open automatically. The application should also be saved in the Applications folder.
- Focus on users’ immediate tasks. Simplify data selection and presentation to display only the information that users need at any one moment.
Introduction to Web based application development for mobile devices

- Minimize the number of times that users need to click the trackwheel, trackball, trackpad, or touch screen to complete a task.
- Design your UI (User Interface) to allow users to change their mind and undo commands. For example, always display a dialog box if users click a destructive menu item or button. Users sometimes click the wrong menu item or button accidentally.
- Display information in a way that makes effective use of the small screen.

Before you design your application, consider using the core applications on the BlackBerry device or the BlackBerry Smartphone Simulator to learn more about the navigation model and best practices for designing the UI.

Create custom application control policies for a BlackBerry Java Application

After you add a BlackBerry Java Application to the application repository, you can configure the application to use the standard application control policies, or you can create custom application control policies for the application. If you want a BlackBerry Java Application to use custom application control policies, you must create the custom application control policies before you add the application to a software configuration. When you add the application to a software configuration, you can select which custom application control policy you want to apply to the application.

If you add the BlackBerry Java Application to multiple software configurations and you assign different custom access control policies to the BlackBerry Java Application in the different software configurations, you must set the priority for the custom application control policies. This priority determines which custom application control policy the BlackBerry Policy Service applies if you assign multiple software configurations to a user account.

1. In the BlackBerry Administration Service, on the BlackBerry solution management menu, expand Software > Applications.
2. Click Manage applications.
3. Search for a BlackBerry Java Application.
4. In the search results, click a BlackBerry Java Application.
5. In the Application versions section, click the version of the application that you want to create a custom application control policy for.
6. Click Edit application.
7. On the Application control policies tab, in the Settings section, select the Use custom application control policies option.
8. Perform any of the following tasks:
9. If necessary, in each section, click the up and down arrows to set the priority for the application control policies.

10. Click Save all.

### Designing web page content for the BlackBerry Browser

Design content so that users can quickly find the content they require.

- Practice minimalism to avoid sending unnecessary content. Web pages should include only what is necessary for the user to complete the intended task. Remove content that does not further the purpose of the web page.
- Place important information at the top of the page where it is most quickly accessible.
• Reduce the vertical space occupied by top-of-page header components, such as banner images and navigation bars. On BlackBerry devices with landscape-oriented screens, large headers can take up valuable screen space.
• Label links carefully, and clearly identify the target of each link, to help users decide which links they want to follow.

Page length

Consider the length of your web pages carefully. The amount of content on a web page has several implications. The more content sent, the fewer requests the user makes, but it can also increase the time it takes to download and render content and the amount of scrolling users must do. Sending less content can result in faster download and rendering times and more focused content, but it can increase the number of HTTP requests the user makes to complete their task.

Number of requests

It is typically better to provide slightly more content on a web page and reduce the number of HTTP requests for which the user must wait.

Use AJAX

To improve download and rendering speeds, you can use JavaScript® and Asynchronous JavaScript and XML (AJAX) requests to progressively render the page. Using AJAX requests can result in an improved user experience, since perceived wait times are reduced. However, to benefit from AJAX, the user must have JavaScript support turned on.

Use dynamic HTML

Consider using dynamic HTML effects to show and hide content dynamically. Using dynamic HTML allows you to place more information on the page without using up extra screen space. For example, you can cluster key headings at the top of the page.

The user can then click a heading to expand a topic of interest to read more. Use dynamic HTML judiciously; however; keep in mind that for your dynamic HTML to be functional, the user must have JavaScript support turned on.

Using fonts effectively in your web content

• Modify fonts judiciously. Use the user-defined default font for the BlackBerry® Browser where possible, unless you have a specific need to select a different font. Although using the user-defined default font makes it more difficult to control layout and appearance, it avoids issues of
choosing a font face or font size that the user finds hard to read. Users will either change the
default BlackBerry Browser font to a suitable font face or and font size, or accept the themespecified default font.

• Avoid absolute font sizes where possible. If you change the font size, use relative sizes, such as
larger, smaller, and xx-large. In standard text flows, such as a series of paragraphs, changing the
font size is unnecessary, because text will wrap. The user can specify the smallest font size for
the BlackBerry Browser, so relative or absolute font sizes that are less than the specified
smallest font size produces no effect on a user's device.

• Verify that the specified font is legible on BlackBerry devices.

• Use CSS to define fonts and font style information. Avoid using the <font> or <basefont>
elements to define fonts for your content. These elements are deprecated. Avoid using HTML
elements such as <bold> or <italic>, to define fonts. Instead, use semantic elements that suggest
the type of information that is enclosed, for example <em>, then set a style for that element
using CSS.

• Use bold fonts for headings and button text. Consider carefully the need for bold fonts elsewhere
in your content. Excessive use of bold fonts for emphasis can be distracting and impair the user's
ability to read your content. Although the BlackBerry Browser supports all values for the CSS
font-weight property, it translates specified values into one of two font weights: normal and
bold.

• Use italics and underlined text judiciously. These font styles can make text hard to read.

• Use colored text carefully. Consider the legibility of text when choosing colors. Text becomes
illegible when it appears on a background of similar color or brightness.

Designing web page layout

With extensive support of web standards such as HTML 4.01, CSS 2.1, JavaScript 1.5, and DOM Level 2,
the BlackBerry Browser can capably render and display most content. Most content, however, is not
optimized for viewing on BlackBerry devices.

The web page layouts that are used for desktop-oriented web content are typically too complex for
small screens. Many web pages use two-column or three-column layouts, with the principle content
placed in one column, and other columns typically devoted to site navigation, advertisements, related
links, and other nonessential elements.

Most standard web content is designed at widths of between 900 and 1100 pixels, while most
BlackBerry devices have screens between 240 and 480 pixels wide. You must decide what content is
essential for the purposes of the web page.

The structural components common to every web page are as follows:

• A page header. The header typically contains branding information such as a logo or banner
image.

• A navigation structure. The navigation is often in a side bar, or in a navigation bar located below
the header.

• The page content.
A page footer. Footers often include secondary navigation links or contact information.

Because of the small screen width on BlackBerry devices, when you create content for the BlackBerry Browser, stacking the structural components vertically is the most effective layout.

**Designing web page layouts for BlackBerry Browser content**

- Keep web page layouts simple. Avoid complex, multicolumn layouts that do not present well on small screens. Instead, stack structural components vertically.
- Use CSS to define web page layouts, where possible. Using CSS allows you flexibility in terms of altering presentation to suit different browsers.
- Avoid using tables to lay out content, where possible. Although you can use tables to define the layout of basic page components, using tables hard codes the layout and reduces your ability to adapt content to different screen sizes. With BlackBerry® Device Software version 4.6 and later, when you display content in the Column View, the BlackBerry Browser ignores table layout and places one cell in each row.
- Design web page components that adapt to different screen widths. Avoid using absolute measures to specify element widths. You can use proportional measures to allow the BlackBerry Browser to size elements proportionally to the screen width.

**Controlling initial web page scaling**

On BlackBerry devices with BlackBerry Device Software version 4.6 or later, by default, the BlackBerry Browser renders desktop-oriented content as it appears on a desktop computer, then scales the content to fit the BlackBerry device screen. For most desktop-oriented content, the BlackBerry Browser renders it at 20 to 40% of its original size.

Content that you design for the BlackBerry Browser must be presented to the user unscaled. You must inform the BlackBerry Browser that content is designed for the BlackBerry device screen by adding a `<meta>` tag to control the initial scale of your web page. Otherwise, the BlackBerry Browser scales your content in error.

The BlackBerry Browser supports the following methods to control the initial display scale with the `<meta>` tag:

- **HandheldFriendly**: The HandheldFriendly processing instruction informs the BlackBerry Browser that the content contained within the document is designed for small screens.
- **viewport**: The viewport definition allows you to specify the specific dimensions of the intended viewport.

Although there are differences in the implementation of each method, you can achieve the same result using either method.
Controlling initial scale using the HandheldFriendly processing instruction

The HandheldFriendly processing instruction informs the BlackBerry Browser that the content is optimized for small screens.

When content the user defines as HandheldFriendly is rendered in Page View, the BlackBerry Browser renders it on a canvas equal to the size of the BlackBerry device screen.

The HandheldFriendly processing instruction is defined in a `<meta>` element within the `<head>` section of an HTML document.

```html
<meta name="HandheldFriendly" content="true" />
```

Controlling initial scale using the viewport definition

You can use the viewport definition to specify the dimensions of the viewport in which the content is designed to appear.

The BlackBerry Browser renders the content on a canvas that is equal to the specified dimensions, then scales it to fit the BlackBerry device screen.

The BlackBerry Browser supports the height and width properties for this tag. You can specify absolute values for these properties, or use the relative device-height and device-width constants. For best results, you can specify device-width as the value of the width property. Specifying the height is typically unnecessary.

The viewport definition is specified in a `<meta>` element within the `<head>` section of an HTML document.

```html
<meta name="viewport" content="width=device-width" />
```

Designing site navigation for the BlackBerry Browser

Effective navigation is vital for BlackBerry Browser content. Users want to quickly find the content they need, with as few wrong turns as possible. Following links erroneously in a wireless environment is costly, both in potential data charges, and in the time spent retrieving and rendering an unwanted web page.

Desktop-oriented browser page navigation structures are not appropriate for small screens.

You can improve site navigation as you plan your web site. Begin by creating an effective site hierarchy. A consistent hierarchy of web pages helps to make your site more predictable and easier to navigate. You can organize your web pages based on their purpose to minimize the number of clicks—and the amount of time—that it takes users to find information or perform tasks. In addition, you should provide a short,
descriptive title for each web page. The BlackBerry Browser lists web pages in the History list by their title. Clear titles can help users retrieve previously visited pages from the History list.

As you create content, within each web page, reduce the primary navigation links to the bare minimum. Commonly used sidebar navigation is not appropriate for display on most BlackBerry device screens. Instead, a simple horizontal navigation bar with a few targeted links in your content header works best. If necessary, consider adding a footer to your content to contain additional navigational links.

In all cases, label links carefully. Clearly identify the target of each link to help users decide which links they want to follow, and to avoid having them follow links in error.

Using JavaScript, you can create dynamic effects such as drop-down menus to provide additional navigation without using additional screen space. However, dynamic navigation structures are not a good web design practice, and are not effective for BlackBerry Browser content that must be accessible by all BlackBerry device users. Consider the following implications:

• Dynamic content relies on JavaScript support being turned on. Navigation should not rely on JavaScript support. If you create dynamic navigation structures, ensure that users can still access at least top level links if JavaScript support is turned off.
• Dynamic content is only supported in BlackBerry Device Software version 4.6 and later. If you create dynamic navigation structures, users with BlackBerry devices running earlier versions of the BlackBerry Device Software cannot access links.

Best practices: designing your web site navigation

• Create an effective site hierarchy. If web pages are organized well, users can intuitively find the information they need.
• Provide a short, descriptive title for the page. The BlackBerry Browser displays the title in a title bar above the content, and uses the title to identify the web page in the History and Go To lists. From the page title, the user must clearly identify the content and purpose of each web page.
• Provide consistent navigation aids. However you choose to present your navigation aids, you must keep them consistent across your web site. Inconsistent navigation aids force users to spend time trying to understand navigation behavior and can cause the user to follow links in error. Consistent navigation structures give your web pages a predictability that helps users navigate through your web site.
• Consider providing primary navigation links in a horizontal navigation bar. Links should appear on a single line at the top of the page.
• Simplify your site navigation. Reduce the number of links on a web page to those that are most relevant.
• Include a link to the home page at the top of every page.
• Consider adding a Search field at the top of the page to allow users to search your web site for content.
• If necessary, include secondary navigation links at the bottom of the page.
• Avoid sidebar navigation. Additional right or left columns are inappropriate for small screens.
• Avoid displaying breadcrumbs at the top of the screen. Although breadcrumbs can help users to orient themselves within your web site and quickly return to previously viewed pages, on small screens, breadcrumbs occupy too much screen space. Look for alternative ways to present this information to the user.
• Be cautious about using dynamic HTML to create drop-down lists for navigation bars. Although you can use these techniques to provide more navigation from the navigation bar without using more screen space, they rely on the user having JavaScript support turned on. Ideally, navigation aids should not rely on user configurable settings.
• Be aware that the BlackBerry Browser does not support dynamic pseudo-classes such as :hover.

Using tables in your BlackBerry Browser content

In BlackBerry Device Software version 4.6 or later, the BlackBerry Browser provides full support for tables. However, you should use tables sparingly in content designed for the BlackBerry Browser. Although tables are effective in providing structure to content, tables can result in presentation problems on small screens, since they often require more space than is available.

Unless the table is small, the entire table might not be viewable unless the user scales the content to fit, in which case it may not be legible, or scroll horizontally or vertically. The result is that the scannability of data, one of the main benefits of tables, is diminished.

You can improve table presentation in content designed for BlackBerry devices by designing tables to fit the width of the screen. If you want tables, you must be aware of the following implications of table presentation in the BlackBerry Browser:

- Screen widths among different BlackBerry devices can vary, from between 240 to 480 pixels. Tabular configurations that work on wide screens can present poorly on narrow screens.
- On BlackBerry devices with BlackBerry Device Software version 4.6 or later, if users view content in the Column View, the BlackBerry Browser ignores the table layout, and instead renders tables vertically with each cell on its own line. As a result, if you include tables, you may be requiring users to change their content view.

You can expect that most users will view content primarily in Page View, and therefore accept that a small number of users will need to change their content view to view table data as intended.

Best practices: using tables

- Minimize the use of tables in your content. Where possible, look for alternative ways to present table data. For example, you can effectively present tabular data in a definition list or other list format.
- Use tables for data presentation only. Avoid using tables to define layout of pages or for page structures such as forms. Use CSS for layout, instead of tables, wherever possible.
- Design tables that do not exceed the screen width. To allow the browser to resize the table to fit different screen sizes, use proportional measures, rather than specifying fixed widths.
• Design tables to use the simplest layout possible. If tables are overly complex, consider whether you can split the information into separate tables. Avoid nested tables.

• Use CSS styles to visually define table structure and improve the scannability of table data. Clearly differentiate header information from data.

• Make tables stylistically minimal. On small screens, excessive styling can cause visual clutter and can make table data harder to scan.

• Minimize the use of borders. Heavy borders can hinder legibility rather than improve it. Defining table structure using background shading and color is typically more effective.

• Consider collapsing borders. By default, table borders are defined with a value of separate, meaning each cell has its own borders, rather than sharing borders with adjacent cells. Collapsed borders remove unnecessary spacing between cells and present a more streamlined appearance. To collapse cells, apply the following style in your style sheet:

\[
\text{border-collapse: collapse;}
\]

**Designing forms for the BlackBerry Browser**

The BlackBerry Browser is well-suited to collecting information from the user using forms. However, the smaller screen sizes and wireless connections of BlackBerry devices mean that the layout and information design of forms takes on greater importance.

Excessive screen clutter, scrolling, or wait times can all impair the user's ability to complete your form and can result in a poor user experience.

In general, well-structured, minimalist layouts work best for forms on BlackBerry devices. Text input fields, selection lists and other form controls should be aligned rather than positioned beside the label to create a consistent and clean presentation.

Forms with proper alignment are easier to read and complete than those with ragged layouts.

To reduce the amount of vertical space required, use a two-column layout, with labels aligned to the right or left in one column, and form controls left aligned in the other column.

If a narrow screen width prevents a manageable two-column layout, a single-column layout is a good alternative, with labels above the form control and all form content left aligned. Single-column layouts require more vertical space, and therefore more scrolling by the user, but the ordered structure of single-column layouts makes them a better choice than ragged layouts in which form controls are not aligned.
<table>
<thead>
<tr>
<th>Layout</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>two column right-aligned labels</td>
<td><strong>Advantages</strong></td>
</tr>
<tr>
<td></td>
<td>• Two-column layouts use less vertical space.</td>
</tr>
<tr>
<td></td>
<td>• Right-aligned labels are clearly associated with the corresponding form controls.</td>
</tr>
<tr>
<td></td>
<td>• The proximity of the labels makes it easy to see the relationship between grouped elements.</td>
</tr>
<tr>
<td></td>
<td><strong>Disadvantages</strong></td>
</tr>
<tr>
<td></td>
<td>• Right-aligned labels are less easily scanned.</td>
</tr>
<tr>
<td></td>
<td>• Right-aligned labels require more horizontal space.</td>
</tr>
<tr>
<td></td>
<td>• Limited label space means labels must be brief.</td>
</tr>
<tr>
<td>two column left-aligned labels</td>
<td><strong>Advantages</strong></td>
</tr>
<tr>
<td></td>
<td>• Two-column layouts use less vertical space.</td>
</tr>
<tr>
<td></td>
<td>• Left-aligned labels are easily scanned.</td>
</tr>
<tr>
<td></td>
<td>• The proximity of the labels makes it easy to see the relationship between grouped elements.</td>
</tr>
<tr>
<td></td>
<td><strong>Disadvantages</strong></td>
</tr>
<tr>
<td></td>
<td>• Left-aligned labels are less clearly associated with the corresponding form controls.</td>
</tr>
<tr>
<td></td>
<td>• Right-aligned labels require more horizontal space.</td>
</tr>
<tr>
<td></td>
<td>• Limited label space means labels must be brief.</td>
</tr>
<tr>
<td>one-column labels aligned above the form control</td>
<td><strong>Advantages</strong></td>
</tr>
<tr>
<td></td>
<td>• Single-column layouts can be processed by the user quickly.</td>
</tr>
<tr>
<td></td>
<td>• Labels are clearly associated with the corresponding form controls.</td>
</tr>
<tr>
<td></td>
<td>• Extra horizontal space for labels makes the form more flexible for complex or localized labels.</td>
</tr>
<tr>
<td></td>
<td><strong>Disadvantages</strong></td>
</tr>
<tr>
<td></td>
<td>• Single-column layouts require more vertical space. As a result, fewer form controls are viewable on the screen, and more scrolling is necessary.</td>
</tr>
<tr>
<td></td>
<td>• Top-aligned labels make it more difficult to identify relationships between grouped elements.</td>
</tr>
</tbody>
</table>
Best practices: creating usable form designs

Form layouts can be simple, structured, and flexible.

- Use a two-column layout where horizontal space permits, with labels in the left column and form controls in the right column. This layout uses less vertical space and is generally better suited to BlackBerry device screens.
- Use a single-column, vertical layout where the screen width prevents a usable two-column layout.
- Use CSS to control form layout, rather than using tables. Because of the variation in screen width on different BlackBerry devices, consider presenting device-specific layouts for narrow and wide screens. Use CSS to provide greater flexibility in presentation.
- Left-align form controls. Aligning form controls, such as text fields or list boxes, improves the readability of the form and helps to reduce visual clutter.
- Space form controls appropriately. Provide enough space between form controls so that a user can easily select a control using the device pointer or a finger.
- Group related form elements together. Organizing elements into groups provides a visual and logical structure to form content, which allows the user to process form content more quickly.
- Minimize the visual impact of your form. Reduce the number of visual elements you use to provide organization. For example, avoid using borders around groups of form elements, where a single horizontal rule separating groups is sufficient. Similarly, if you use a background color to group content, keep it subtle.

Adding form controls

Best practices: adding text input fields

- You can create text input fields using the `<input type="text">`, `<input type="password">`, or `<textarea>` elements.
- Provide fields that are sized appropriately for the data they are intended to accept. The size of a field provides a visual cue to the user about the input required. For example, if the user is required to enter a ZIP code, create a field that is five characters wide.
- Where the length of text input is unknown, use a consistent field width. Any `<textarea>` elements should conform to this width.
- Define a border for all text input fields. Users must click the field to activate it and enter text. Borders clearly define the clickable area for the user.
- Provide a default value for text input fields, where possible. Default text provides a cognitive cue to the user about the input required. For example, you can use a default value to provide hint text, provide a brief description of the kind of information the field is intended to collect, or assist with field completion (such as providing HTTP:// in a field intended to collect a URL).
• Define an input mask for text input fields, where possible. The BlackBerry Browser supports the -wap-input-format WAP CSS extension. This CSS style allows you to define the kind of input the field can accept. Define input masks to provide field level input validation without requiring scripts.

Best practices: Adding labels to form controls

• Enclose text for form control labels in the <label> element, and define the for attribute to associate the label and the corresponding form control. This represents well-structured, accessible HTML, which offers more versatility when defining CSS presentation properties that you can use with screen reader technology.
• Provide clear, concise label text.
• Use title case capitalization for labels.
• Punctuate labels for form controls using a colon (:).
• Identify optional or required form controls. If your form contains a mix of optional and required form fields, you must communicate this to the user.
• Indicate whichever is least prevalent, the required or optional elements. That is, if you have fewer required fields than optional fields, indicate the required fields. Where required and optional fields are relatively evenly split, indicate the required fields. Identifying the smaller number of fields helps to reduce visual clutter.
• Use text or a symbol such as an asterisk (*) as an indicator. Associate the indicator with the label that corresponds to the form control. Text provides greater clarity. However, to save space, symbols are preferable on BlackBerry devices.
• Consider using color to highlight special-case indicators.
• Provide a legend to describe what the indicator symbol denotes. If you use a color symbol, use the same color for the legend text to visually associate the legend and the symbol.

Best practices: Adding buttons

You can create buttons using the <input type="submit">, <input type="reset">, or <button> elements. Buttons allow users to perform actions such as submit the form, move to a new screen, or save form data.

• Add buttons judiciously. Typically, each page should have a Submit button or a button that moves users forward through the completion process, and a Save button. You can add a button to allow users to move back to previous screens, or a Reset or Cancel button. Before you add other buttons, consider whether they are necessary and can be helpful to the user.
• Present the buttons for secondary or destructive actions, such as the cancel function, in a manner that is visually distinct from, and less conspicuous, than buttons for primary actions, such as the submit or save function. By creating a visual differentiation, you draw the user’s attention away from negative actions and allow primary action buttons to have greater focus.
• Include a Save button on every form. Adding a Save button helps the user save unsubmitted or partially completed form data if they cannot complete the form in one continuous block of time.
• Use one-word labels for buttons, where possible. A descriptive verb that identifies the action the button performs is ideal. For example, Save, Submit, Cancel, Reset.
• Limit the use of custom images for buttons. In general, text labels are more reliable than images. Images work best when they are universally understood symbols. For example, in a long form, you can use arrow images for buttons that allow the user to navigate back and forth through form screens.
• Position buttons appropriately based on their importance. The primary action button, typically Submit or Continue, should be aligned at the left edge of the form control. Aligning the primary button with other form controls creates a visual path from the top of the form to the point of completion.
• Avoid using punctuation with buttons.

Creating wireless-friendly forms

Although forms generally work well in the BlackBerry Browser, there are several characteristics of the BlackBerry browsing environment to consider as you develop content.

A loss of network coverage can be a problem for users trying to complete and submit forms. Unless the user keeps the browser session open, input can be lost. On wireless devices, loss of network connectivity can occur more frequently than on desktop computers. BlackBerry devices can lose network connectivity for a number of reasons, for example, if battery power is too low, or the user moves into an area without wireless network coverage.

With slow network speeds, there are design choices you can make to minimize the impact of the network, particularly when lengthy forms are required. If users complete a form page, and then must request additional form content from the server, the time it takes to complete and submit the form increases, and can result in a poor user experience.

Consider the following guidelines to mitigate the impact of the wireless network:
• Validate form input on the device, where possible, to prevent wasting time and bandwidth transferring data that is invalid or unusable.
• Define form submission queues. To reduce the impact of loss of wireless network connectivity, set up form submission queues to allow the BlackBerry Browser to queue a form submission request, and then submit the form data when the device regains a network connection.
• Provide a Save button. Allow users to save the form input if they are unable to complete it.
• Reduce the number of HTTP requests the user must make and for which they must wait for responses. Look for ways to download and display form content that minimizes wait times. For example, you can use AJAX requests to retrieve content in the background and progressively display it as it is received. Multiple HTTP requests are still required, but because data is retrieved as the user completes the form, wait times can be reduced.
Creating form submission queues

To reduce the impact of loss of wireless network connectivity, the BlackBerry Browser allows you to set up form submission queues. By configuring form submission queues, users can still complete and submit several forms (possibly for different queues), even if their BlackBerry device is outside of a wireless coverage area.

When the user attempts to submit a form without being connected to the wireless network, the BlackBerry Browser queues the submit requests until the network connection is reestablished. When the BlackBerry device returns to a wireless coverage area, the browser submits its queued forms in the order in which they were queued, without any user intervention. After the queued forms are submitted, user responses are stored by the browser in the same queue list. Users can open the queue list, and then click the form queue to view the response. The BlackBerry Browser only queues submitted form data when the BlackBerry device is outside a wireless coverage area.

You can create form submission queues in the following ways:

- Create an HTTP property file: Creating an HTTP property file in which you define the queuing parameters lets you create and manage multiple form queues in a single location. You must set your web server to send the headers when the web page that contains the form is requested. You can add queuing parameters for additional forms within the same header file.

For example, the following Apache Tomcat™ configuration file defines a queue for a form on a web page named mypage.xhtml:

```xml
<Files mypage.xhtml>
  Header set cache-control max-age=2592000
  Header set x-rim-queue-id Register
  Header set x-rim-request-title "Stock Monitor"
  Header set x-rim-next-target success.xhtml
</Files>
```

Add the queuing parameters directly to the HTML content. In HTML or XHTML, queuing parameters are added using hidden `<input>` elements.

```html
<input type="hidden" name="x-rim-queue-id" value="Register" />
<input type="hidden" name="x-rim-request-title" value="Stock Monitor" />
```
### Form submission queue parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Required</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>x-rim-queue-id</td>
<td>yes</td>
<td>text_string</td>
<td>This parameter specifies the offline form queue to which the BlackBerry Browser sends any GET or POST requests from the web page when the BlackBerry device is outside a wireless coverage area.</td>
</tr>
<tr>
<td>x-rim-next-target</td>
<td>no</td>
<td>text_string</td>
<td>This parameter specifies the next web page to send after sending any GET or POST requests to the form queue.</td>
</tr>
<tr>
<td>x-rim-request-title</td>
<td>no</td>
<td>text_string</td>
<td>This parameter specifies the label used to identify this request in the Queue view page. By default, the title of the page is used to identify the request.</td>
</tr>
<tr>
<td>x-rim-request-id</td>
<td>no</td>
<td>True</td>
<td>False</td>
</tr>
<tr>
<td>x-rim-request-date</td>
<td>no</td>
<td>True</td>
<td>False</td>
</tr>
</tbody>
</table>

### Creating and using images

Images are often used in web content with no meaningful effect. In wireless web content, reduced bandwidth of wireless networks means that you must use images judiciously and optimize images for...
wireless browsing or risk creating a poor user experience. For example, large images require considerable bandwidth, can render slowly, and can result in higher costs to the user.

When a BlackBerry device requests a web page through the BlackBerry Mobile Data System (MDS) Connection Service or the BlackBerry Internet Service (BIS) Browsing network gateway, images are automatically optimized for rendering on the BlackBerry device. The network gateways compress images and deliver them in segments to the BlackBerry Browser. On the BlackBerry device, the browser renders images progressively, so that content flow can be rendered before the entire image is downloaded.

The BlackBerry MDS Connection Service is an optimized application development framework for the BlackBerry Enterprise Solution. It reduces the amount of time and resources required to develop and deploy wireless applications for mobile workers. BlackBerry MDS allows organizations to deliver corporate data wirelessly, leveraging the same push delivery model and advanced security features used for BlackBerry email.

The BlackBerry Internet Service Browsing network gateway is an email and Internet service for BlackBerry devices that is designed to provide subscribers with automatic delivery of email messages, mobile access to email message attachments, and convenient access to Internet content.

Not all BlackBerry device users will access your content through these network gateways. As a result, you must optimize images before you send them to BlackBerry devices.

You can detect the BlackBerry device model that is requesting your content and send images that are optimized for the screen dimensions of the BlackBerry device. Sending device-appropriate images reduces the volume of data that is transferred to the BlackBerry device and allows you to ensure you use images to their best effect for each screen size.

**Best practices: Using images**

- Use images judiciously. Ensure each image you add to your web page adds value and meaning.

- Scale images appropriately for the screen size before sending them to BlackBerry devices. Because of the variance in screen width among different BlackBerry device models, sending images optimized for a particular screen size is ideal, where possible.

- Crop images to provide a meaningful impact on small screens. Details visible when an image appears on a desktop computer can often be lost when that content is scaled for display on a BlackBerry device.

By cropping images to remove unnecessary detail while retaining meaningful information, you increase the value of the image to the user.

- Avoid using text in images. Text in images can become illegible if the image is scaled, and text that is embedded as part of the image cannot be localized.

- Use the alt attribute to provide context when users do not download images, or for users who are sight-impaired. If an image is purely decorative, and there is no need for alternative text, provide a null value (""").
• Specify the dimensions of the image. When the user accesses content over WAP or Wi-Fi® connections, specifying image dimensions allows the BlackBerry Browser to determine how text flows around the image without requiring it to load each image first. When the user accesses content through the BlackBerry MDS Connection Service or BlackBerry Internet Service Browsing network gateways, specifying image dimensions is typically unnecessary, since those network gateways can specify image dimensions for the BlackBerry Browser while preprocessing the content. You can specify image dimensions in your style sheet, or you can use the height and width attributes of the `<img>` tag.

• Beware of using icons to represent navigational links. You can provide icon-based navigation to reduce the space required for links; however, if the user cannot easily identify what the icon represents, icons can reduce clarity. Providing clear text-based links is typically a safer option.

• If you decide to use icons, limit their use to easily identifiable images. Consider using icons similar to those used in popular web content, desktop applications, or operating systems, with which users are likely to be familiar.

<table>
<thead>
<tr>
<th><strong>Best practice</strong></th>
<th><strong>Description</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>Make JavaScript files external.</td>
<td>Linking to external JavaScript files requires an additional HTTP request, which might initially increase the time to retrieve and load the web page. However, because external JavaScript files are cached, over time, linking to external JavaScript files can result in time savings. Inline scripts must be loaded every time the page is retrieved, whereas external script files result in time savings, since if the script file is cached this front-end cost is outweighed by the savings experienced. To reduce the impact of first-time visits, you can inline a subset of JavaScript on the first page, and load the JavaScript after the page is loaded. Load time is reduced for the Home page, but other web pages can reference the cached script resources.</td>
</tr>
<tr>
<td>Use the <code>&lt;noscript&gt;</code> element.</td>
<td>The user can turn JavaScript support on the BlackBerry device on or off, or it can be turned on or off by an IT policy. You can use the <code>&lt;noscript&gt;</code> element to ensure that your content is still usable after JavaScript support is turned off. The <code>&lt;noscript&gt;</code> element defines an alternate block of HTML content to be executed if the BlackBerry Browser is unable to run the script.</td>
</tr>
</tbody>
</table>
Avoid blocking page rendering unnecessarily.

When the BlackBerry Browser parses a web page, the browser processes JavaScript as it is encountered and page rendering is blocked. The browser must first determine if the script affects the initial page content. If the script does not affect the initial page content, then blocking page rendering is an unnecessary delay.

Consider the following guidelines to avoid unnecessarily blocking page rendering:

Separate the scripts that affect page content from those that do not.

Load scripts at the end of the page, rather than from within the `<head>` element, or use the `defer` attribute for scripts that do not affect page content.

```
<script defer src="http://www.your_company.com/js/my_script.js">
```

Avoid using `eval()` functions to execute JavaScript code.

Using `eval()` functions to execute JavaScript code can negatively affect performance.

The BlackBerry MDS Connection Service network gateway and the BlackBerry Internet Service Browsing network gateway precompile and compress scripts before sending them to the BlackBerry Browser. The BlackBerry Browser does not compile script as quickly as the network gateways. Precompiling the script can reduce the time it takes to execute the script.

The network gateways do not evaluate the content of the JavaScript file. As a result, the contents of `eval()` method is treated as a string. If the method contains JavaScript code, the code is passed to the BlackBerry Browser uncompiled, and the browser must compile the code, increasing the time it takes to execute the script.
Avoid calling additional resources from within scripts.

Unless the requests are asynchronous, requesting additional resources from within a script can cause delays for the user.

Typically, the BlackBerry MDS Connection Service network gateway and the BlackBerry Internet Service Browsing network gateway speed up resource retrieval by requesting resources on behalf of the BlackBerry Browser as they process the content. However, the network gateways do not evaluate the contents of scripts; they merely precompile the code and then send it to the BlackBerry Browser.

If you request a resource such as another JavaScript file from within a script, the BlackBerry Browser must make an HTTP request over the wireless network to retrieve the resource, and pause the execution of the script until the resource is retrieved.
BlackBerry Web Development Plug-in for Eclipse 1.0

The BlackBerry® Web Development Plug-in for Eclipse® is a new addition to the web development tooling products. This product is seamlessly integrated with Eclipse® 3.4. It allows you to debug browser-based applications and content for the BlackBerry device, profile the code, and increase efficiency from using Eclipse in a familiar manner. The Eclipse plug-in also facilitates remote web development, debugging and profiling. Included with the plug-in is a BlackBerry Smartphone Simulator to assist you in immediately being able to test content.

New Web Development Enhancements

Debugging

• Support for debugging web projects with the BlackBerry Smartphone Simulators
• Setting Breakpoints on JavaScript code
• Seamless integration into Debug, Variables and Expressions views

Profiling

• Visibility into the content of XMLHttpRequest requests and response data
• Visibility into data traffic for web-based content, including elements such as images, CSS, JavaScript, HTML (Hypertext Markup Language)
• Reporting on time-to-load for web-based content, including elements such as images, CSS, JavaScript, HTML

Simulator

• BlackBerry® Bold™ smartphone simulator now included as the default simulator for web projects in the BlackBerry® Integrated Development Environment IDE package

Requirements

• Operating System: Windows® XP and Windows Vista™ (32-bit)
• Eclipse: Eclipse 3.4.1, EMF 2.4.1, WTP 3.0.3
• Java: Java 1.6
• Installing Eclipse Updates: North American users can use the update site to download and install the components directly into an Eclipse install. For more information about available components, add the following URL into the Eclipse update manager: www.blackberry.com/go/eclipseUpdate.
Applications designed for BlackBerry devices should provide features that are simple to use, and preserve resources. When you design your BlackBerry device application, consider the differences between mobile devices and computers.

Mobile device users use applications on their mobile device differently than they use applications on a computer. On mobile devices, users expect to find information quickly.

When you design applications for BlackBerry devices, be as consistent as possible with other BlackBerry device applications.

After you add a BlackBerry Java Application to the application repository, you can configure the application to use the standard application control policies, or you can create custom application control policies for the application.

Design content so that users can quickly find the content they require. When designing web page layouts, tables, and forms, use best practices to optimize the presentation.

If web pages are organized well, users can intuitively find the information they need.

Use tables for data presentation only. Avoid using tables to define layout of pages or for page structures such as forms.

Validate form input on the device, where possible, to prevent wasting time and bandwidth transferring data that is invalid or unusable.

Ensure each image you add to your web page adds value and meaning.

The BlackBerry® Web Development Plug-in for Eclipse® allows you to debug browser-based applications and content for the BlackBerry device, profile the code, and increase efficiency from using Eclipse in a familiar manner. The Eclipse plug-in also facilitates remote web development, debugging and profiling. Included with the plug-in is a BlackBerry Smartphone Simulator to assist you in immediately being able to test content.
Review Questions

1. Why is it a good practice to use the user-defined font in your web pages. What practices should you avoid in using fonts even when using the user-defined font?

2. What is the benefit of adding this tag to your web page

   <meta name="HandheldFriendly" content="true" />

   What is the benefit of using this tag and how does this differ from using the previous tag?

   <meta name="viewport" content="width=device-width" />

3. Describe best practices in table layout, focusing on how to use Tables effectively and what alternatives you have to using Tables.

4. Describe the problems with using large images on wireless devices and how those problems can be addressed.

5. Describe the benefits and costs of embedding JavaScript in the web page as opposed to using external JavaScript files. How can these differences be used to reduce download time?