

Nokia WRT Plug-in for Aptana Studio QuickStart Guide

Version 1.0; 18 December 2008

WRT

NOKIA

Copyright © 2008 Nokia Corporation. All rights reserved.

Nokia and Forum Nokia are trademarks or registered trademarks of Nokia Corporation. Java and all Java-based marks are trademarks or registered trademarks of Sun Microsystems, Inc. Bluetooth is a registered trademark of Bluetooth SIG, Inc. Other product and company names mentioned herein may be trademarks or trade names of their respective owners.

Disclaimer

The information in this document is provided 'as is', with no warranties whatsoever, including any warranty of merchantability, fitness for any particular purpose, or any warranty otherwise arising out of any proposal, specification, or sample. This document is provided for informational purposes only.

Nokia Corporation disclaims all liability, including liability for infringement of any proprietary rights, relating to implementation of information presented in this document. Nokia Corporation does not warrant or represent that such use will not infringe such rights.

Nokia Corporation retains the right to make changes to this document at any time, without notice.

Licence

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

Contents

1	Introduction.....	5
2	Installing the plug-in.....	6
3	Creating a new widget project	7
4	Overview of widget projects	8
5	Editing and code completion	9
6	Previewing a widget	10
6.1	Previewing within Aptana Studio.....	10
6.2	Previewing in an external browser	13
6.3	Previewing in an S60 emulator	14
7	Debugging a widget	15
8	Widget packaging	16
9	Widget deployment.....	18
9.1	Additional information for emulator deployment.....	19
10	Evaluate this resource	21

Change history

18 December 2008	Version 1.0	Initial document release
------------------	-------------	--------------------------

1 Introduction

This document provides a quick-start overview to using the Nokia WRT Plug-in for Aptana Studio. No specific knowledge of Aptana Studio is assumed in this guide, so it should be useful to both new and existing Aptana Studio users.

The guide covers:

- Installing the plug-in.
- Creating a new widget project.
- An overview of widget projects.
- Editing and code completion.
- Previewing a widget.
- Debugging a widget within Aptana Studio or Firefox (using Firebug).
- Widget packaging.
- Widget deployment via Bluetooth connectivity.

This guide assumes that you have already installed Aptana Studio or Aptana Studio Pro 1.2 or later. The current version can be obtained from the [download page on the Aptana website](#).

2 Installing the plug-in

The Nokia WRT Plug-in is installed from within Aptana Studio. To do this, with Aptana Studio open, select **Help > My Aptana > Plugins** as shown in Figure 1. The Nokia WRT Plug-in is listed under the **Platforms** tab and can be installed via the **Get it** link.



Figure 1: The Nokia WRT Plug-in is available for installation from the My Aptana feature in Aptana Studio.

Once the **Search Results** dialogue opens, select the item for the site providing the Nokia WRT Plug-in before selecting **Next**. The licence agreement now is displayed; choose **I accept the terms of the license agreement** before clicking **Next**. The **Installation** dialogue now displays the items that will be installed. Click **Finish** to download the plug-in. Once the download is complete, the **Feature Verification** dialogue is displayed. Select **Install all** to start the installation.

Once the plug-in has been installed, Aptana Studio checks for updates when it is started, once every 24 hours it is running, and any time a manual update check is performed.

3 Creating a new widget project

To create a new WRT widget project, select **File > New > Project** from the Aptana Studio menu. The **Select a wizard** dialogue is now displayed. Open the **Web Runtime (S60)** item to see the options:

- **Import Web Runtime Widget (S60)** — Use this option for an existing widget project that you want to import into Aptana Studio. Please note that widget installation files (WGZ) will need to be unpacked before they can be imported.
- **New Web Runtime Widget (S60)** — Use this to create a new project. You will be given the option to use one of three widget templates: Basic Widget Project, Basic Widget Project with WRTKit, or RSS Reader Widget Project.

4 Overview of widget projects

WRT widget projects are standard Aptana web projects, with some special content, as shown in Figure 2.

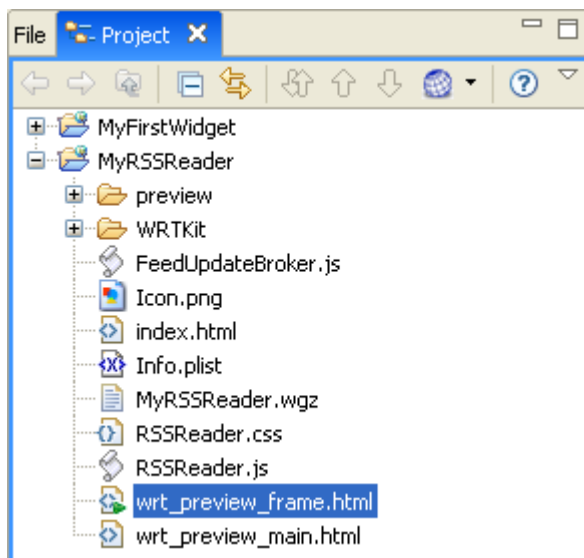


Figure 2: The content of a typical WRT widget project in Aptana Studio is shown.

The content of the project is as follows:

- *Preview folder* — This folder is added to all widget projects by the plug-in. It contains the support files needed to preview a widget in a desktop browser. You should not edit these files.
- *WRTKit* — This folder contains the WRTKit library. This is an optional library that can be used to help create attractive, WRT-compatible widgets quickly. If you do not use the WRTKit files in a widget, the folder should be deleted or excluded from the final widget package. For more information on excluding files, see Chapter 8, ‘Widget packaging’.
- *wrt_preview_frame.html* — This file is generated by the plug-in. It facilitates the previewing and debugging of the widget. You should not edit this file. For more information, see Chapter 6, ‘Previewing a widget’, and Chapter 9, ‘Widget deployment’.
- *wrt_preview_main.html* — This file is generated by the plug-in. It is a copy of the user’s main HTML file (the one listed in `info.plist`), modified to include the JavaScript™ files necessary for previewing the widget in Aptana Studio. You should not open or edit this file. For more information, see Chapter 6, ‘Previewing a widget’.

5 Editing and code completion

Aptana Studio provides the HTML, Cascading Style Sheets (CSS), and JavaScript editors required for the creation of widgets. When you edit JavaScript code, Aptana Studio provides completion for web standards and popular desktop-browser JavaScript extensions. The WRT plug-in installs code-completion support for the Web Runtime 1.0 API.

When JavaScript code is being edited, this feature activates when typing begins in the JavaScript editor, as shown in Figure 3, or in a script element of the HTML editor.

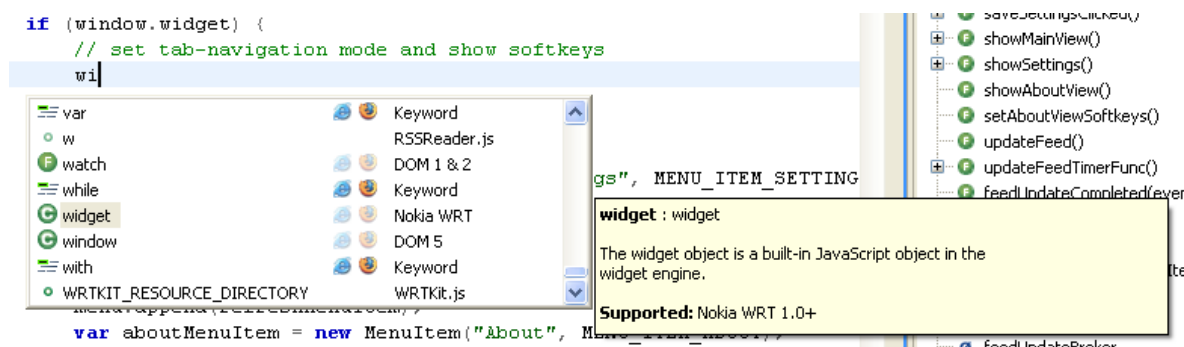


Figure 3: Code completion, with extended API information, installed by the plug-in is shown.

All Web Runtime 1.0 API objects, properties, and methods are supported. The documentation text is the same as that presented in the Forum Nokia [Web Developer's Library](#). The library can be accessed from **Help > Help Contents** within Aptana Studio and is available from the [Documents/Examples](#) section of the Forum Nokia website.

Note: The plug-in includes the Web Developer's Library v1.3. The WRT APIs documented in this version of the library are later versions than the API information available through code completion (which is limited to Web Runtime 1.0 API 1.0). Therefore, not all valid WRT APIs will be supported with code completion.

6 Previewing a widget

6.1 Previewing within Aptana Studio

Aptana Studio provides a multitab HTML editor, with built-in preview. From the project tree, double-clicking an HTML file (such as `index.html`, as shown in Figure 2) opens the file in the editor. At the bottom of the editor pane are several tab items, as shown in Figure 4.

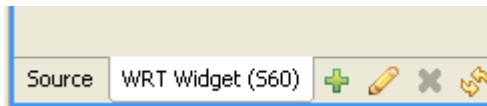


Figure 4: The WRT Widget (S60) tab in the HTML editor is shown.

The **Source** tab is the syntax-aware HTML editor. Clicking the **WRT Widget (S60)** tab provides a preview of the widget in Aptana.

The widget tab configures the web browser to display `wrt_preview_frame.html`, which provides the device-like GUI around the widget content, as shown in Figure 5. The preview offers a resolution selector at the top right of the preview window and a display-orientation switch directly above the widget GUI.

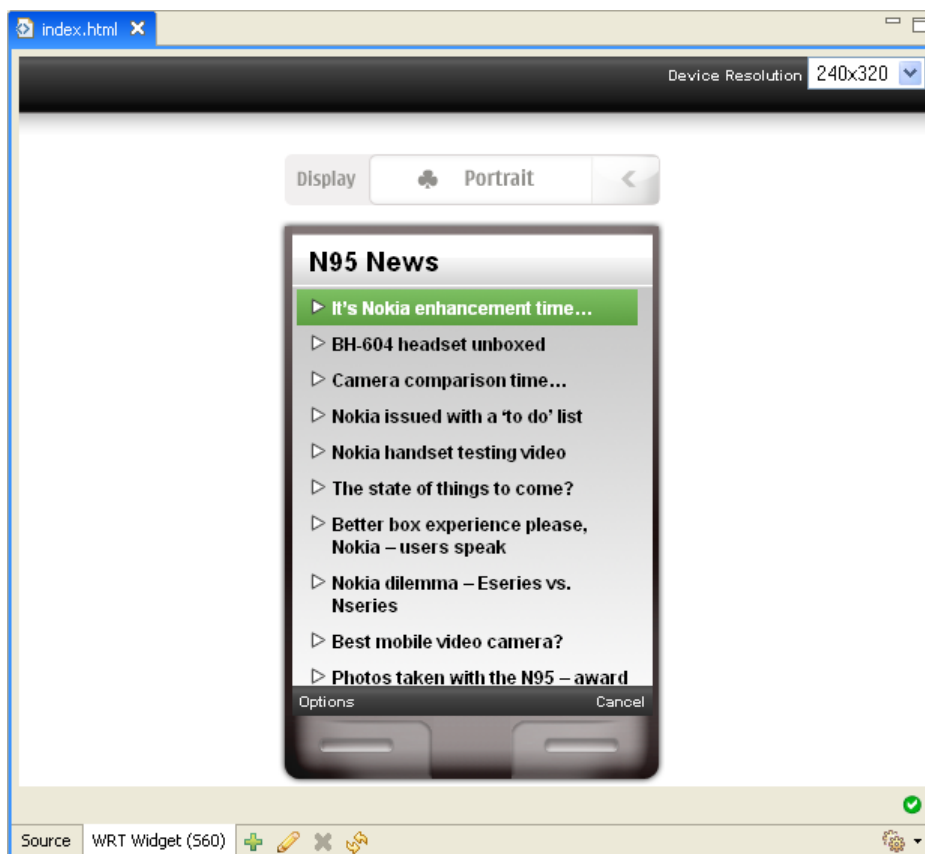


Figure 5: The RSS-feed widget is shown in the Aptana Studio preview.

Note: By default, Aptana Studio uses Firefox to provide the preview. Firefox is based on a different rendering engine from the one used in the Web Browser for S60, which is based on WebKit. As a result, minor differences in rendering may be noticed between the look of a widget in the preview and on an S60 device. In addition, the rendering of fonts will be dependent on the fonts installed on your computer.

You can configure the browser used in the **WRT Widget (S60)** preview tab, by selecting **Window > Preferences** from the Aptana Studio menu. In the preferences dialogue is a page called **Web Runtime** with an item called **Preview**, as shown in Figure 6. On Apple Macintosh computers, the browser choices are Firefox and Safari.

When this preference is changed, any open editors must be closed and re-opened for the change to take effect.

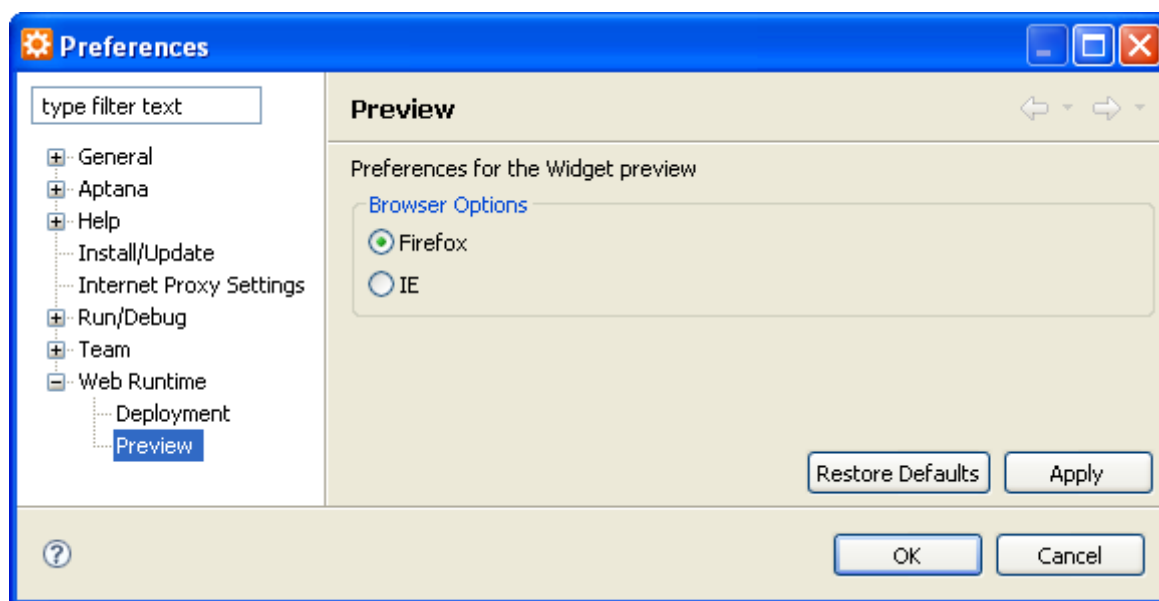


Figure 6: The Widget Preview option in Preferences, allowing the preview browser to be selected, is shown.

If you are using Firefox as the preview browser and are behind a firewall, then you need to configure the Internet Proxy Settings in Aptana Studio. This is because Aptana Studio uses its own embedded version of Firefox, which is unaware of any other proxy configurations.

To configure the proxy settings, choose **Window > Preferences > Internet Proxy Settings**, as shown in Figure 7. Then enter the appropriate settings for your network.

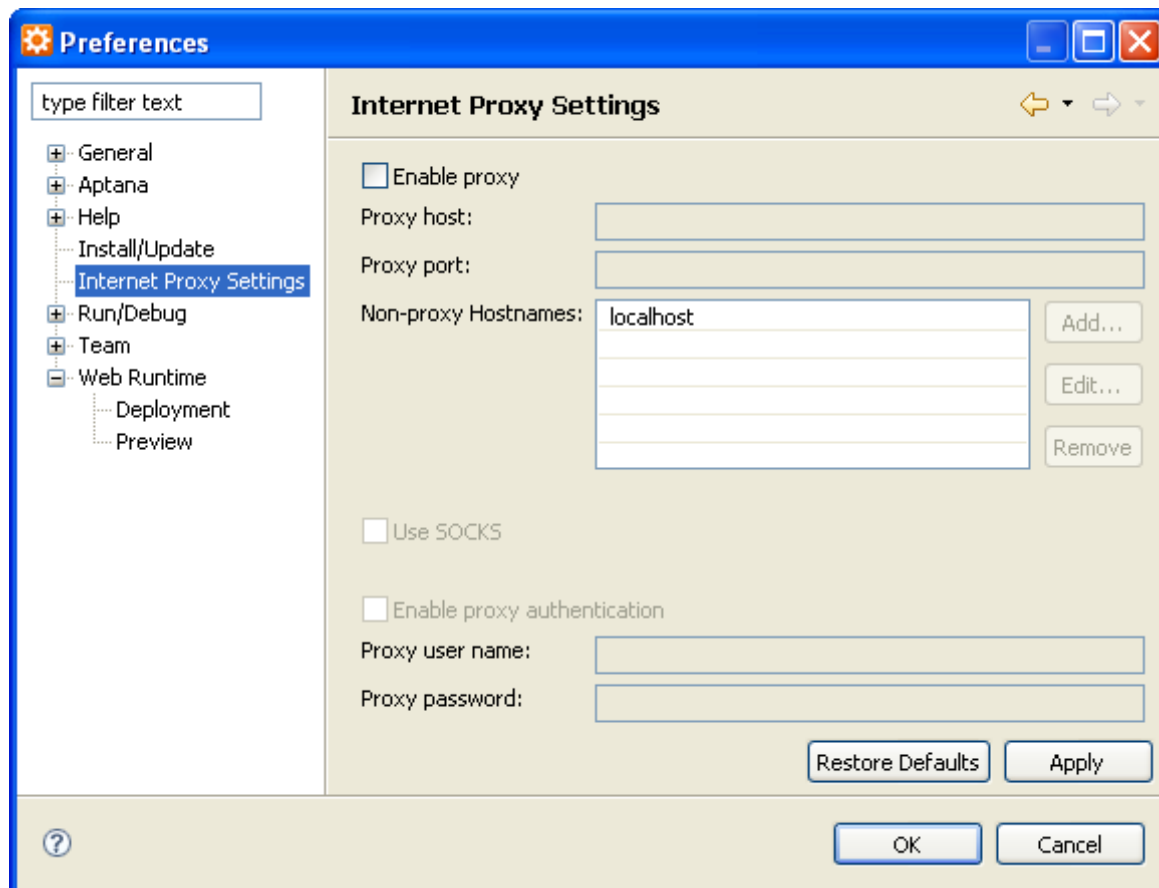


Figure 7: The Internet Proxy Settings preferences page is shown.

In addition, widgets will display a security prompt window when they make an XMLHttpRequest Object. To avoid the display of this message, include the following JavaScript before each XMLHttpRequest.open function:

```
// create XmlHttpRequest object
var xmlhttp = new XMLHttpRequest();

// Enable universal browser read
try{ netscape.security.PrivilegeManager.enablePrivilege
    ("UniversalBrowserRead"); }
catch(e) { }

// before each
xmlhttp.open(METHOD, URL);
```

Users of Apple Macintosh computers must also:

1. Open the Firefox browser and navigate to the URL **about:config**.
2. Search the properties using the filter **signed.applets.codebase_principal_support** and set the Boolean value to true.

For more information, see the articles [How to fix Ajax Error: uncaught exception: Permission denied to call method XMLHttpRequest.open](#) on the Mac OS X Leopard & Tiger Dual Boot blog ([installingcats.com](#)); [Cross-Site XMLHttpRequest](#) on John Resig's blog ([ejohn.org](#)); [HTTP access control](#)

on the Mozilla Developer Center website (developer.mozilla.org); or [Use the XMLHttpRequest Object to Post Data](#) on Murali S.' blog (muralis.wordpress.com).

6.2 Previewing in an external browser

You can also preview widgets in an external browser. This will be of most use to developers working on Windows PCs, because it allows a WebKit-based browser, such as Safari or Google Chrome, to be used for the preview. Previewing with a WebKit based browser should offer the most accurate rendering of the widget. However, it should be noted that debugging is possible only in Firefox.

To preview in an external browser, select the `wrt_preview_frame.html` file and open its shortcut menu. From the shortcut menu, select **Run As > Run**, as shown in Figure 8.

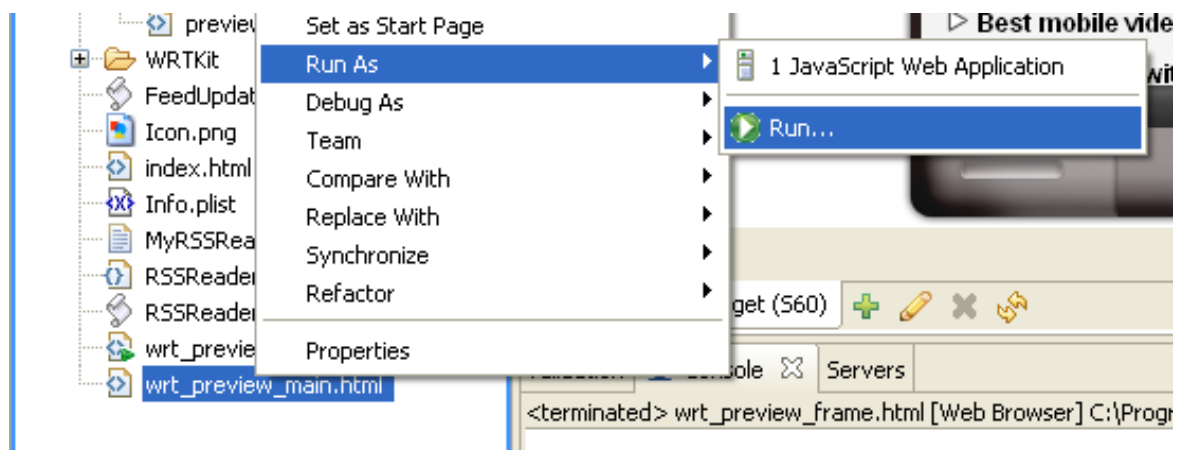


Figure 8: Select Run to enable the preview of a widget in an external browser.

The target browser's executable is then defined in the **Browser executable** field within the **wrt_preview_frame.html** item of the **Web Browser** options in the **Create, manage, and run configurations** dialogue, as shown in Figure 9. Clicking **Run** then launches the browser and starts the preview.

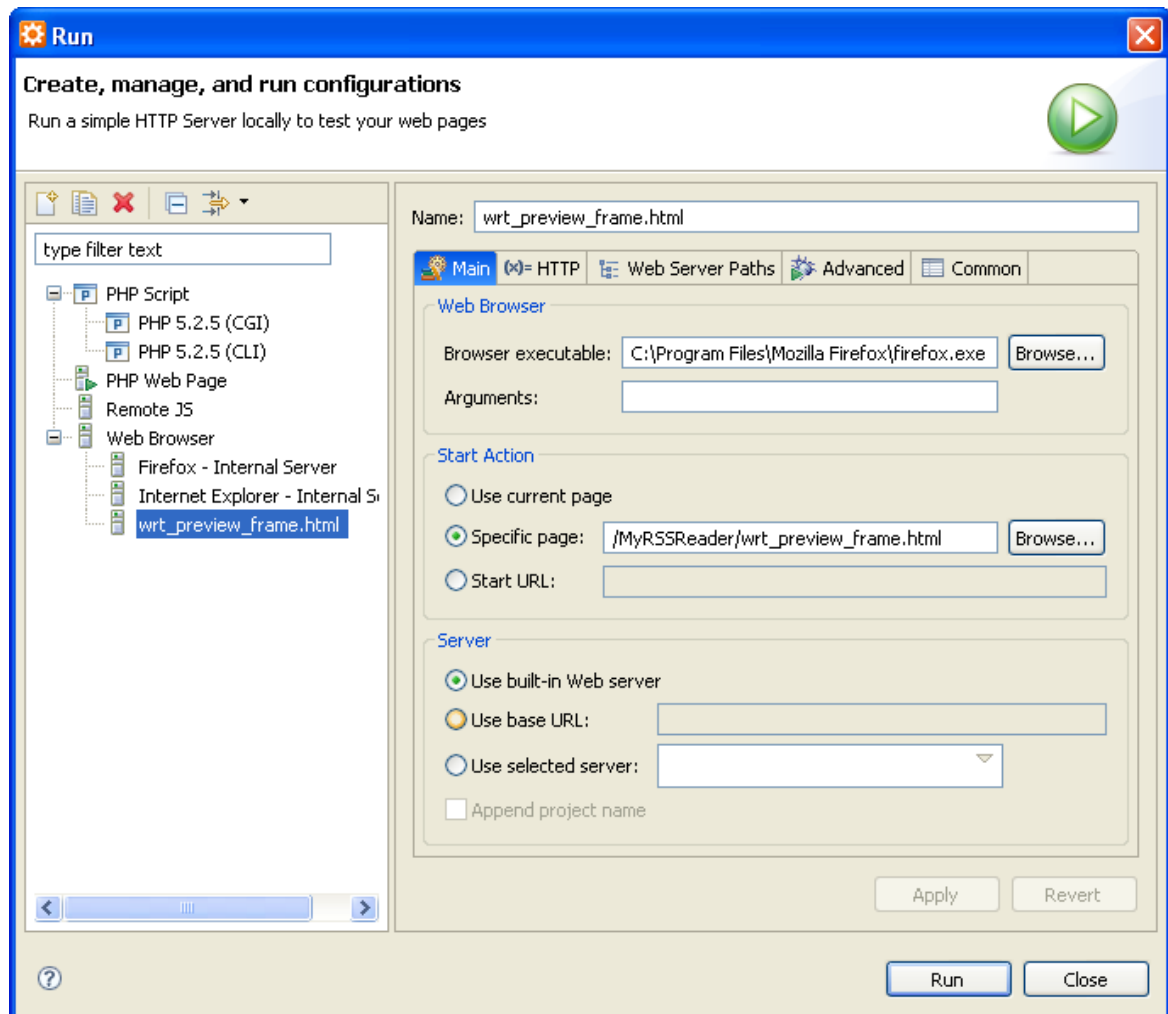


Figure 9: The browser executable is defined as the target for an external preview.

6.3 Previewing in an S60 emulator

It is also possible to preview widgets in the emulators supplied with the [S60 3rd Edition, Feature Pack 2 and S60 5th Edition SDKs](#). To be previewed in this way, widgets need to be packaged (see Chapter 8, 'Widget packaging') and deployed to the emulator (see Chapter 9, 'Widget deployment').

This option is possible only on Windows-based PCs; S60 emulators are not available for the Macintosh OS.

7 Debugging a widget

Debugging a WRT widget uses Aptana Studio's standard debugging facility. However, when you are debugging the WRT API, the emulation environment must be present. Because of this, you must initiate debugging with `wrt_preview_frame.html`, not with the widget's main HTML file.

To start a debugging session, activate the shortcut menu on the `wrt_preview.html` file and select **Debug As > JavaScript Web Application**, as shown in Figure 10. Firefox will be launched as an external application. The first time this is done, Aptana Studio will try to configure Firefox. This may not work. If it does not work, see the Aptana Studio documentation for advice on troubleshooting.

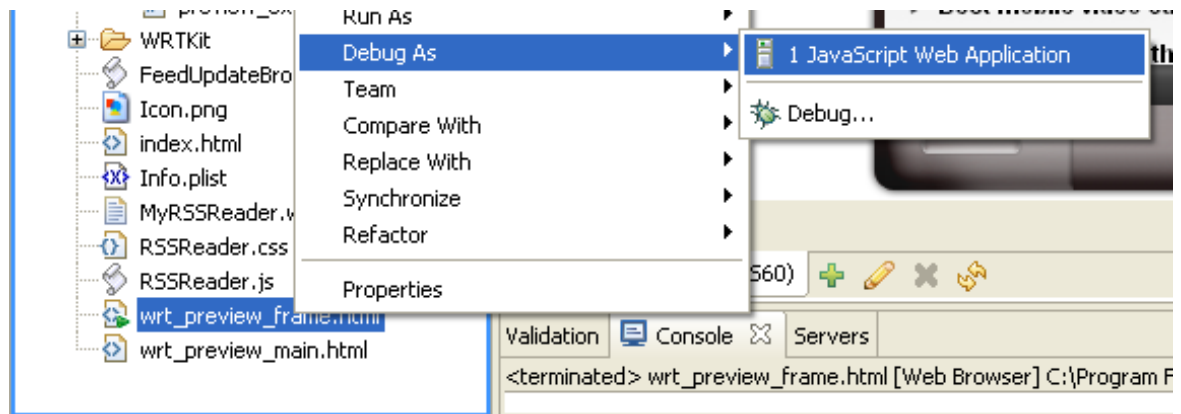


Figure 10: The steps for starting a debugging session from the shortcut menu are shown.

8 Widget packaging

To package a WRT widget project into a *.wgz installation package, open the project's context menu and select **Package Widget**, as shown in Figure 11.

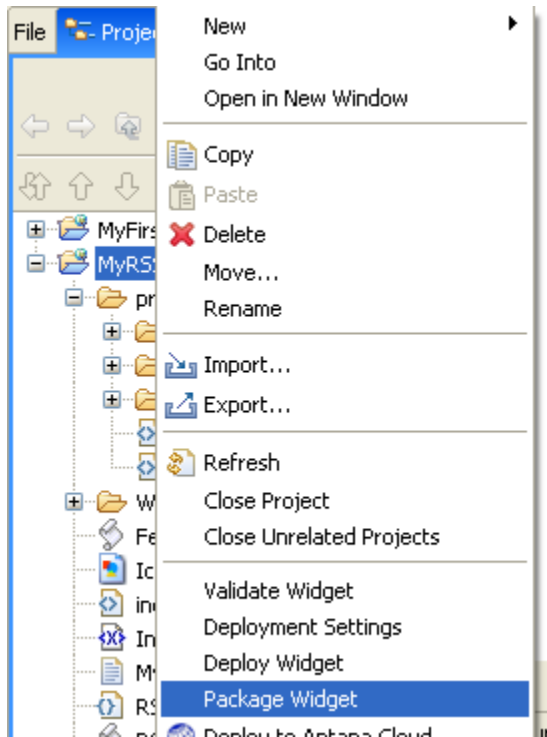


Figure 11: Select Package Widget from the project's shortcut menu.

The *.wgz file is then generated at the top level of the selected project.

By default, all files in the project except for the automatically generated preview support files are added to the installation file. However, it is recommended that you exclude files that are not needed to run the widget on a device, such as documentation files, from the package. Exclude a file from packaging by activating its shortcut menu and selecting **Exclude From Widget Archive**, as shown in Figure 12. To include an excluded file, select **Include in Widget Archive** from its context menu.

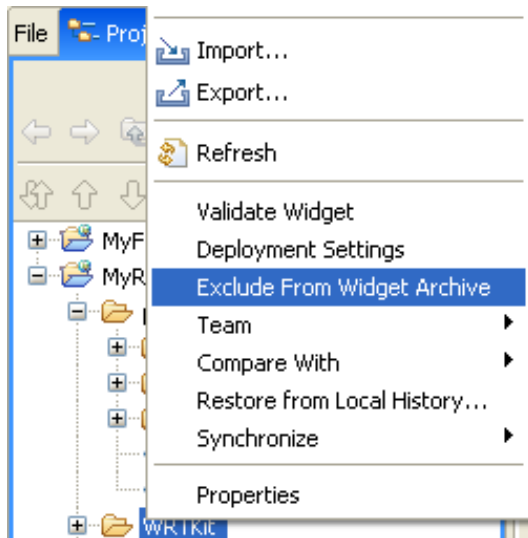


Figure 12: Select Exclude From Widget Archive from a file's shortcut menu.

9 Widget deployment

The plug-in supports the deployment of *.wgz files to S60 devices over a Bluetooth connection using OBEX file transfer or to an S60 emulator (on Windows PCs only). Before this feature can be used, it is necessary to select a deployment target. To do this, select **Window > Preferences** from the Aptana Studio menu, then select the **Web Runtime > Deployment** panel. It is also possible to use the **Deployment Settings** option on the project's shortcut menu, as shown in Figure 13.

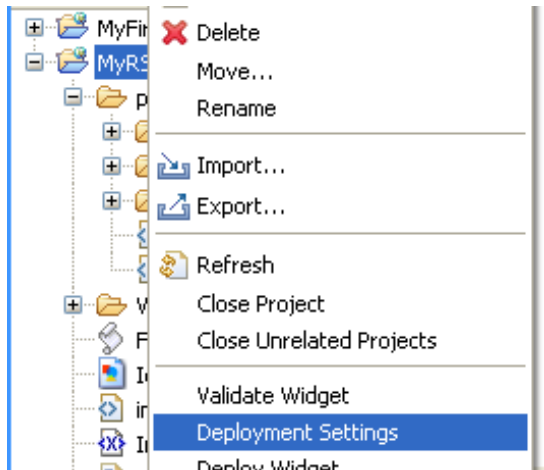


Figure 13: Use the project's shortcut menu to select Deployment Settings.

On Windows PCs there are options to deploy to a device or, where one is installed, to an S60 SDK. On Macintosh computers only device deployment is available.

In the preferences pane, click **Search** to scan for available Bluetooth connections, as shown in Figure 14. When scanning is complete, click **OK** to select a device and confirm the selection.

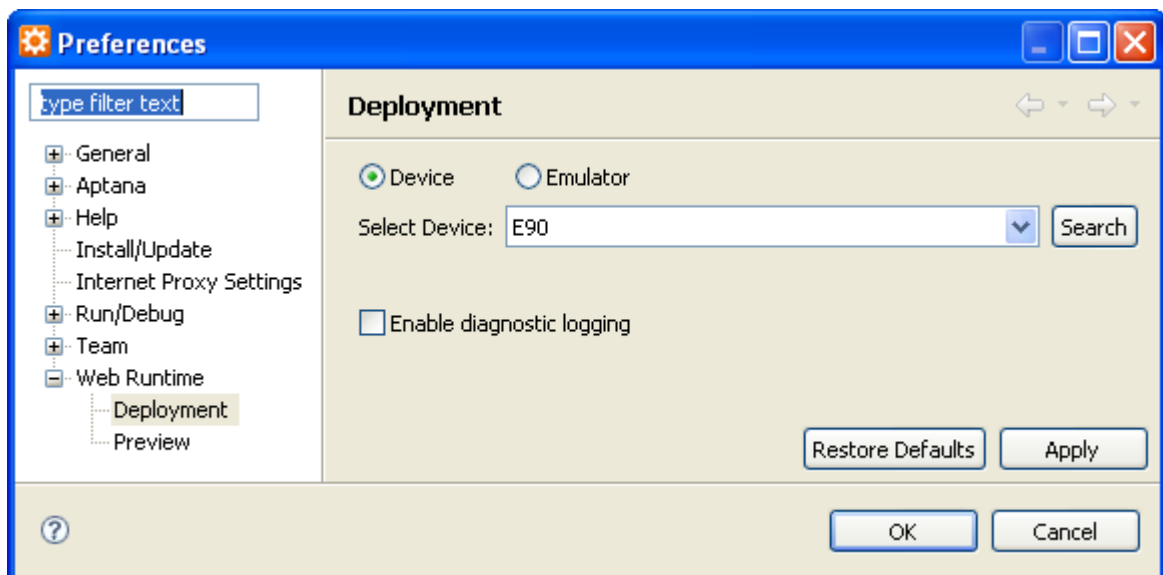
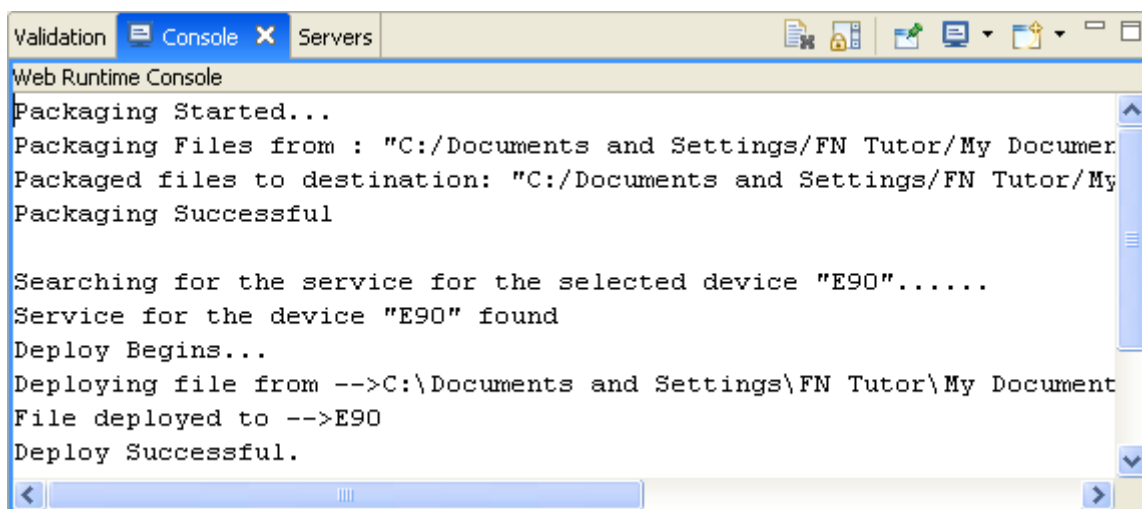


Figure 14: Select a Bluetooth device in the Deployment preferences.

Now, to deploy a widget, open the *.wgz file's or the project's shortcut menu and select **Deploy Widget**. If a project was selected and the project's widget has not yet been packaged, the widget is automatically packaged. If there is an existing archive, that file is used as is.

During the deployment process, status information is listed to a Console window, as shown in Figure 15.



```

Validation Console Servers
Web Runtime Console
Packaging Started...
Packaging Files from : "C:/Documents and Settings/FN Tutor/My Document
Packaged files to destination: "C:/Documents and Settings/FN Tutor/My
Packaging Successful

Searching for the service for the selected device "E90".....
Service for the device "E90" found
Deploy Begins...
Deploying file from -->C:\Documents and Settings\FN Tutor\My Document
File deployed to -->E90
Deploy Successful.
  
```

Figure 15: The Console window shows deployment progress.

9.1 Additional information for emulator deployment

It is possible to deploy WRT widgets to [S60 3rd Edition SDK for Symbian OS, Feature Pack 2 v1.1](#) or [S60 5th Edition SDK for Symbian OS v0.9](#) or later SDKs. Instructions for installing the SDKs are provided in a document included in the download file. These instructions should be followed in their entirety, including the installation of [ActiveState ActivePerl 5.6.1 build 635](#) (later or earlier versions are not compatible).

Once installed and after Aptana Studio has been restarted, the available SDKs are listed in the selector shown in Figure 14 when you select the **Emulator** radio button.

You must then start the emulator before you attempt to deploy a widget from Aptana Studio. The emulator can be started from the **Start** menu.

Note: When installation of a widget is completed, an 'Application closed' error is displayed in the emulator, as shown in Figure 16. This is a known issue and has no effect on the widget installation. The widget will run correctly in the emulator.

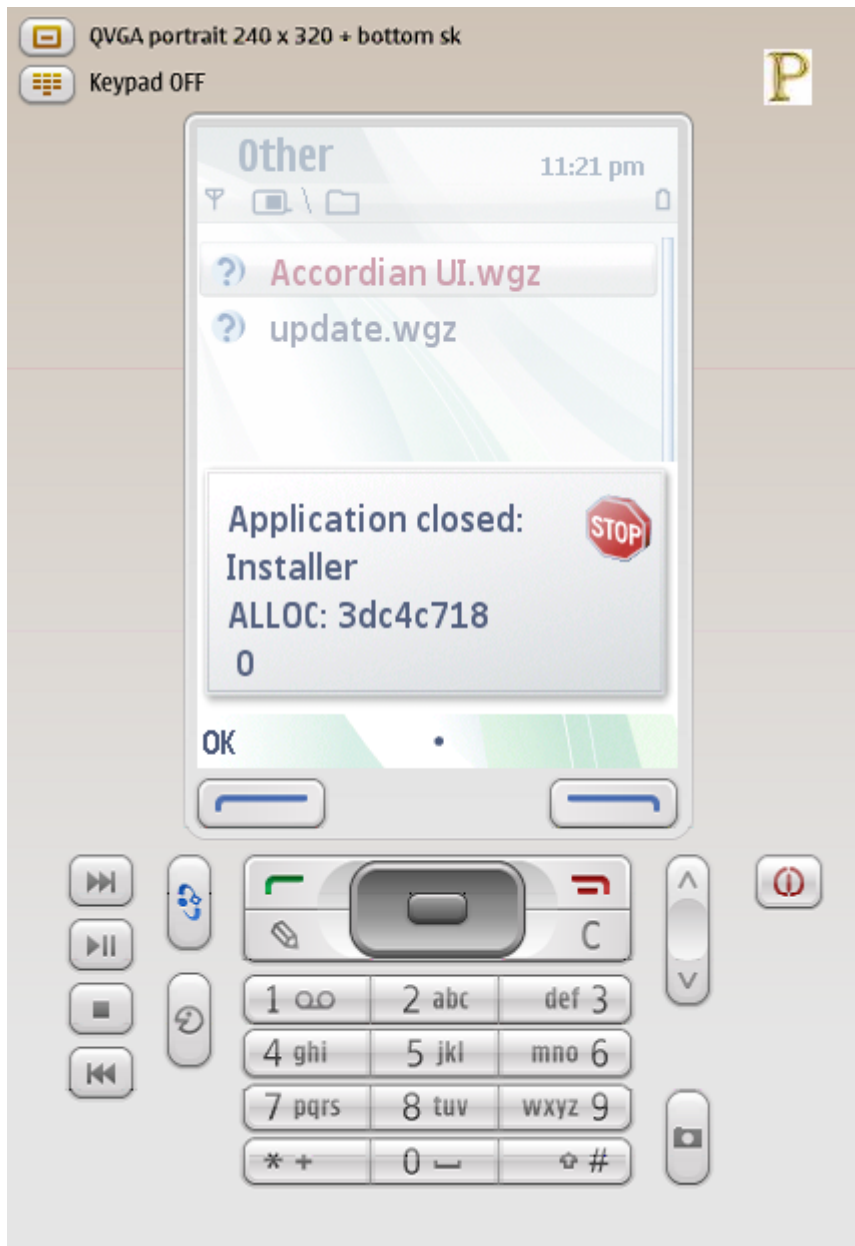


Figure 16: The error that displays after a widget is installed is shown.

10 Evaluate this resource

Please spare a moment to help us improve documentation quality and recognize the resources you find most valuable, by [rating this resource](#).