Oracle® Retail Point-of-Service

Receipt Builder Tool User Guide

Release 14.1

July 2015

ORACLE®

Note: The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle's products remains at the sole discretion of Oracle.

Contents

List of Figures

List of Tables

Preface

This guide describes the Oracle Retail Point-of-Service Receipt Builder Tool user interface. It provides step-by-step instructions to complete most tasks that can be performed through the user interface.

It also describes how to install the Oracle Retail Point-of-Service Receipt Builder Tool.

Audience

This document is intended for implementers who need to change the format or content of receipts and Point-of-Service reports.

Related Documents

For more information, see the following documents in the Oracle Retail Point-of-Service Release 14.1 documentation set:

- *Oracle Retail Point-of-Service Release Notes*
- *Oracle Retail Point-of-Service Installation Guide*
- *Oracle Retail Point-of-Service User Guide*
- *Oracle Retail POS Suite Configuration Guide*
- *Oracle Retail POS Suite Implementation Guide - Volume 1, Implementation Solutions*

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
italic	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
`monospace`	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

1

Installation of the Receipt Builder Tool

Note: The resource detailed in this document is designed to assist customers and implementers in effective use and management of their Retail Point-of-Service applications. Although provided as extended guidance by Oracle Development, the tool/toolkit is NOT a licensed Generally Available (GA) product. As such, in accordance with Oracle Software Technical Support Policies, Oracle Support may not troubleshoot or provide fixes for bugs/issues that might exist or that may be discovered through customer tool usage. Please see official product documentation for technologies that are fully supported.

The Receipt Builder Tool is an XML editor provided as a plug-in to Eclipse. It is used to create and edit receipt blueprint files. For more information, see "Overview" in Chapter 2.

To install the Receipt Builder Tool, follow the instructions in the following sections:

- "Download and Install Eclipse Version 4.3"
- "Obtain JVM Version 7"
- "Verify the Installation of Oracle Retail Point-of-Service"
- "Install the Receipt Builder Tool"

Check the Software Requirements

The following software is required:

- Oracle Retail Point-of-Service Receipt Builder Tool for Release 14.1
- Oracle Retail Point-of-Service Release 14.1
- Eclipse version 4.3, Eclipse IDE for Java Developers package
- Java Virtual Machine (JVM) 7

Download and Install Eclipse Version 4.3

Note: The Receipt Builder Tool has been tested with Eclipse version 4.3, Eclipse IDE for Java Developers package. Other versions, such as EasyEclipse, do not support the Object Inspector plug-in required for the Receipt Builder Tool.

The download is available at the following web site:

http://archive.eclipse.org/eclipse/downloads/drops4/R-4.3-201306052000/

To download and install Eclipse, follow the instructions to download the zip file, for example, `eclipse-SDK-4.3-win32.zip`.

Obtain JVM Version 7

Eclipse must be run with a JVM version of 7. If you do not have version 7 installed, you can obtain the JVM from the Oracle web site. The download is available at the following web site. Follow the instructions at the web site for downloading the correct JRE version.

http://www.oracle.com/technetwork/java/javase/downloads/index.html

Set Up Eclipse to Use the JVM

It is necessary to set up Eclipse to run with the JVM version 7. The JRE may be part of an installed JDK. In the following example, the line that is added to ensure Eclipse launches with a JRE 7 is shown in bold.

```
-startup
plugins/org.eclipse.equinox.launcher_1.1.0.v20100507.jar
--launcher.library
plugins/org.eclipse.equinox.launcher.win32.win32.x86_1.1.1.R36x_v20100810
-showsplash
org.eclipse.platform
--launcher.XXMaxPermSize
256m
--launcher.defaultAction
openFile
-vmargs
-Dosgi.requiredJavaVersion=1.7
-Xms256m
-Xmx512m
```

For more information on setting up the .ini file, see the following web site:

http://wiki.eclipse.org/Eclipse.ini#Specifying_the_JVM

Best Practices for Eclipse Settings

Set Eclipse to use a version 7 JRE as the default JRE and for the compile settings:

1. In the Eclipse window, select **Window** and then **Preferences**. The Preferences window is displayed.
2. Select **Java** and then **Compiler**. For the Compiler Compliance Level, select 1.7 from the menu.

3. Select **Java** and then **Installed JREs**. Check the box for the JRE version 1.7.
4. To save the settings, click **OK**.

Verify the Installation of Oracle Retail Point-of-Service

For information on installing Oracle Retail Point-of-Service Release 14.1, see the *Oracle Retail Point-of-Service Installation Guide*.

Install the Receipt Builder Tool

To install the Receipt Builder Tool:

1. Obtain the `oss-plugins-eclipse-14.1.zip` file from My Oracle Support. Contact Customer Support for access to the zip file.
2. Ensure the obtained zip file contains the following plug-in jars:
   - `oracle.eclipse.receiptbldr_1.3.2.jar`
   - `oracle.eclipse.common_1.0.1.jar`
   - `oracle.eclipse.objinspect_1.2.1.jar`
   - `oracle.eclipse.radix_1.0.1.jar`

   Note: The `oracle.eclipse.radix_1.0.1.jar` is optional. This jar contains a reference chart of integers in various radixes for informational purposes only. It is not needed by the Receipt Builder Tool.
3. Extract the `oss-plugins-eclipse-14.1.zip` file to the *<Eclipse_install_directory>*`/plugins` directory.
4. Restart Eclipse.

2

Using the Receipt Builder Tool

This chapter describes how to use the Receipt Builder Tool to create and update blueprint files.

Overview

The structure and content of receipts and Point-of-Service reports are defined in XML files known as blueprint files. There is one blueprint file defined for each receipt and report. A blueprint file is also available for each receipt and Point-of-Service report in each supported language.

The Receipt Builder Tool provides an editor for visually customizing and building receipts. You can format the content of receipts by dragging and dropping business object attributes into selected areas, as well as using the buttons available on the screen. You can display the receipt output to validate the layout of the receipt for printing.

The available receipt blueprint files serve as a template layout for receipts printed for every transaction in Point-of-Service. You can modify and customize all receipt blueprint files. For a list of the available blueprint files and a description of purpose of each file, see Appendix A. For a description of the XSD for blueprint files, see the *Oracle Retail POS Suite Implementation Guide, Volume 1 - Implementation Solutions*.

For a description of the terms used in this guide, see the Glossary.

Multiple Language Support

For each receipt and Point-of-Service report, a blueprint file is provided for each supported language. Table 2–1 lists the supported languages and file name formats.

Table 2–1 Naming of Blueprint Files for Each Supported Language

Language	Blueprint File Name Format	Example
Chinese (Simplified)	*_zh.bpt	SaleReceipt_zh.bpt
Chinese (Traditional)	*_zh_TW.bpt	SaleReceipt_zh_TW.bpt
Croatian	*_hr.bpt	SaleReceipt_hr.bpt
Dutch	*_nl.bpt	SaleReceipt_nl.bpt
English	*.bpt	SaleReceipt.bpt
French	*_fr.bpt	SaleReceipt_fr.bpt
German	*_de.bpt	SaleReceipt_de.bpt
Greek	*_el.bpt	SaleReceipt_el.bpt

Table 2–1 (Cont.) Naming of Blueprint Files for Each Supported Language

Language	Blueprint File Name Format	Example
Hungarian	*_hu.bpt	SaleReceipt_hu.bpt
Italian	*_it.bpt	SaleReceipt_it.bpt
Japanese	*_ja.bpt	SaleReceipt_ja.bpt
Korean	*_ko.bpt	SaleReceipt_ko.bpt
Polish	*_pl.bpt	SaleReceipt_pl.bpt
Portuguese (Brazilian)	*_pt.bpt	SaleReceipt_pt.bpt
Russian	*_ru.bpt	SaleReceipt_ru.bpt
Spanish	*_es.bpt	SaleReceipt_es.bpt
Swedish	*_sv.bpt	SaleReceipt_sv.bpt
Turkish	*_tr.bpt	SaleReceipt_tr.bpt

Note: If you are using multiple languages, any changes you make to a receipt must be made in the blueprint file for each language you are using.

Create a Java Project

You need to create a project for using the Receipt Builder Tool. To create a Java project:

1. In the Eclipse window, select **File**, **New**, and then **Java Project**. The New Java Project window is displayed.
2. Enter the project name, `<project>`.
3. Click **Create project from existing source**. Set the directory field to the location of the Point-of-Service client install, `<OracleRetailStore>/Client`.
4. Click **Finish**. The classpath for the project should include all jars found in its subdirectories. The blueprint files are available under `<project>/pos/receipts`.

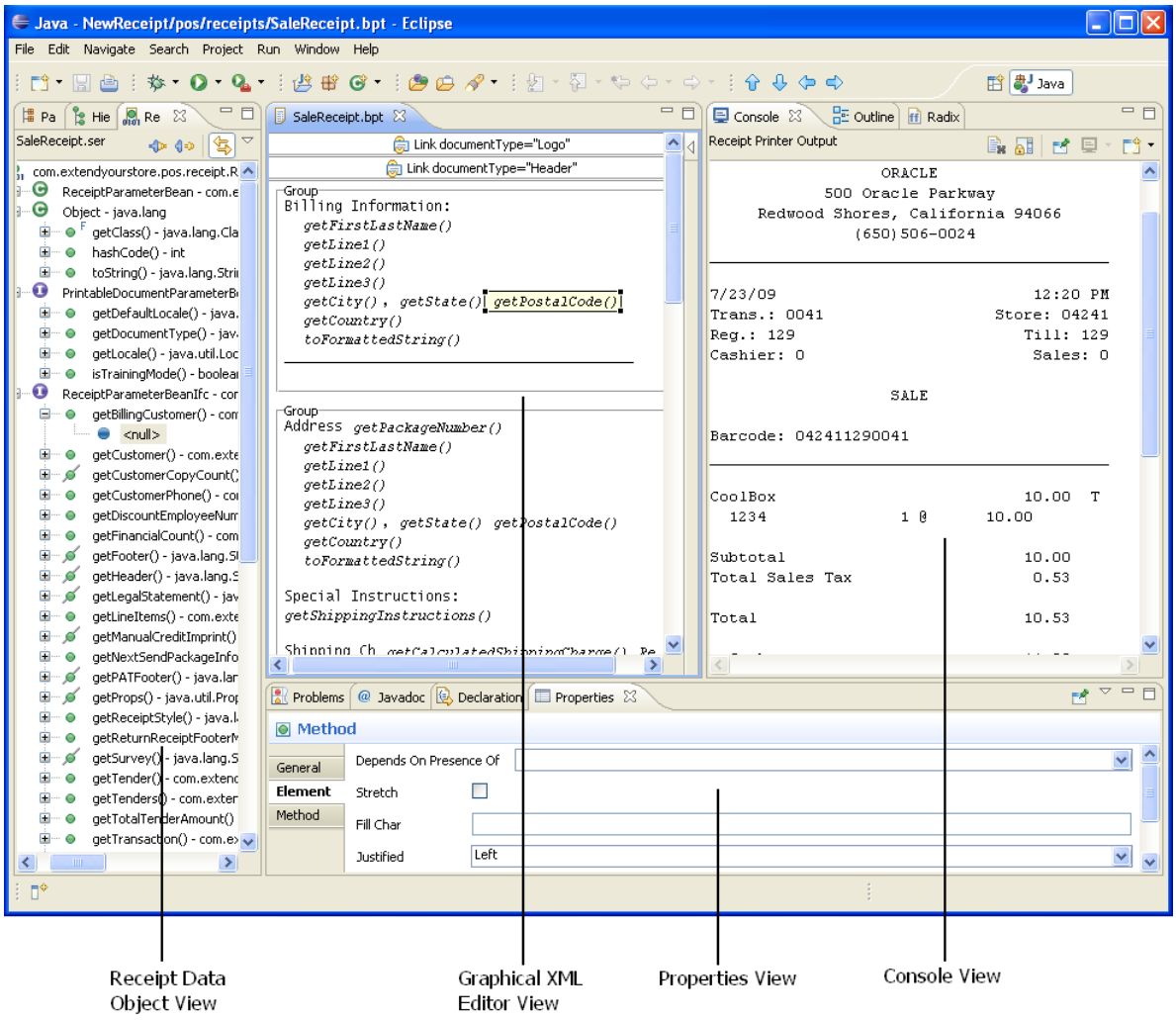
   Additional blueprint files created by the retailer should be added to `<project>/pos/receipts`.

Note: Point-of-Service libraries must exist in the classpath for the Eclipse project. All additional source must be compiled for Receipt Builder to work.

Receipt Builder Tool Window

Figure 2–1 shows an example of the Eclipse window running the Receipt Builder Tool using the Java perspective.

Figure 2–1 Example of the Receipt Builder Tool Window



The views are the working areas of the Receipt Builder Tool. Each view graphically displays the layout and attributes of receipt blueprints, Receipt Data Objects (RDO), and receipt elements. The following views are available:

- Console view—displays the receipt output based upon the attached RDO formatted by the current blueprint when the receipt is printed. Any logo or barcode does not show up in this view.
- Graphical XML Editor view—where you edit the content of the blueprint file. The fonts used in this view can be configured using window preferences. See "Setting up Preferences".

  A palette is available in the Graphical XML Editor view that enables you to add elements to the blueprint file. The flyout palette is opened by clicking the Show Palette button on the right. The elements that can be added are described in Table 2–3.
- The Receipt Data Object view—shows the method structure of the Receipt Data Object currently attached for use with editing the blueprint. If the Receipt Data Object view is closed, it can be opened by selecting **Window**, **Show View...**, **Other**, **Receipt Builder**, and then **Receipt Data Object**.

  - If no Receipt Builder editors are open, opening an editor also opens the Receipt Data Object view.
  - When the last Receipt Builder editor is closed, the Receipt Data Object view is also closed.

  A menu is available in the Receipt Data Object view that enables you to perform different functions in the view. The menu is opened by clicking the View Menu button on the right. For information on the functions, see "Using the Receipt Data Object View".

- Properties view—where you change the properties of the element selected in the Graphical XML Editor view. The Properties view only displays properties for the currently selected element. Properties can also be viewed by selecting the element in the Outline view.

  The choices available in the lists in the Properties view can be configured using window preferences. See "Setting up Preferences".

- Outline view—shows the content of the blueprint file in an outline format.
- Hierarchal Outline view—shows the structure of the XML file and highlights the currently selected element.

Additional views can be opened by selecting **Window**, **Show View...**, and then the view you want to open.

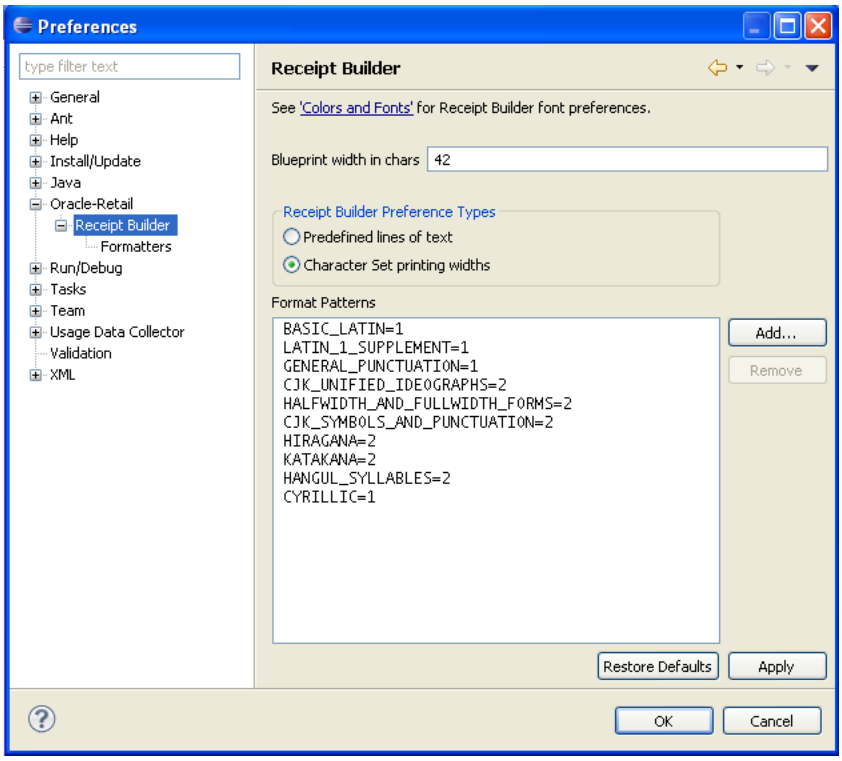
When setting up your Eclipse window, you may want to take the following into consideration :

- Use a secondary perspective, like Resource or Java, from the perspective which you normally use and configure the sash pane sizing of the views similar to Figure 2–1. This enables you to easily maintain and switch back to your primary document editing perspective. Perspectives in Eclipse can be switched by using the buttons on the toolbar or by selecting **Window**, **Open Perspective**, and then choosing the perspective.
- The Console view may work better on the side where it can be squeezed to fit only 42 character width lines. Views can be rearranged in Eclipse by dragging and dropping the tabs.

Setting up Preferences

To select preferences for the Receipt Builder Tool, select **Window** and then **Preferences**. In the Preferences window, expand **Oracle-Retail** and then select **Receipt Builder**. The Receipt Builder preferences are displayed. See Figure 2–2.

Figure 2–2 Preferences Window for Receipt Builder



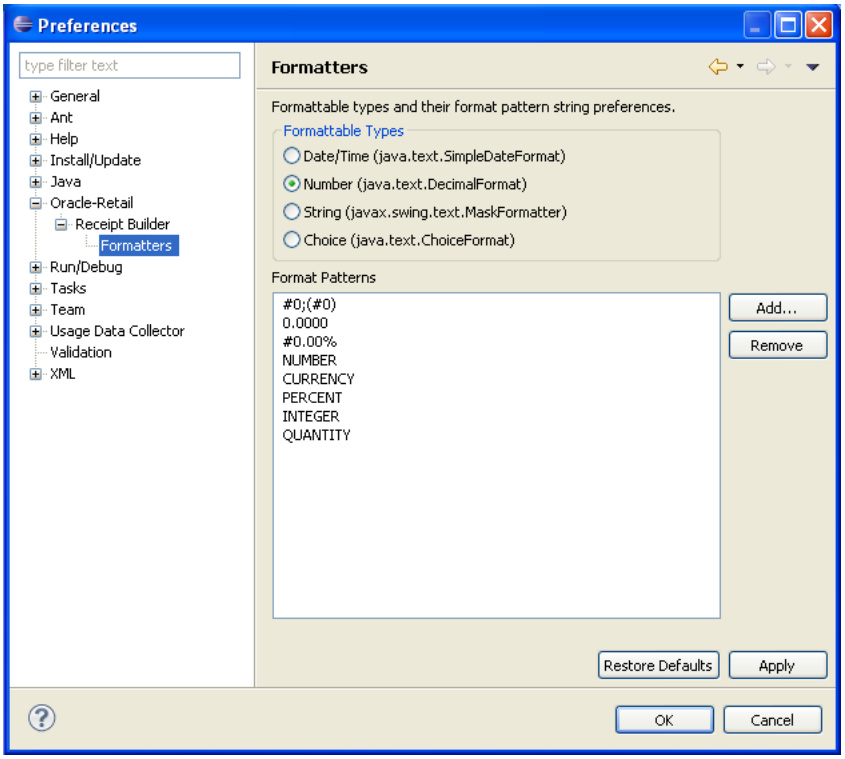
You can change the following formatting:

- You can configure the color and fonts of the information displayed in the Graphical XML Editor view. To configure the colors or font:
  1. Select **Colors and Fonts**. The Colors and Fonts window is displayed.
  2. Expand **Receipt Builder**.
  3. Select the font you want to change and click **Edit...**. The Font window is displayed.
  4. Make your changes to the font and click **OK**. You are returned to the Colors and Fonts window.
  5. When you have completed all your changes, click **OK**.
- *Blueprint width in chars* defines the character width of the output produced in the Console view when printing from the Receipt Builder editor. Enter a new width.
- *Predefined lines of text* sets the values available for the Text list in the Properties view. Change the lines of text in the Format Patterns box:
  - To add a format pattern, click **Add...**. The New Predefined Text window is displayed. Enter the new text and click **OK**.
  - To remove a format pattern, select the pattern in the Format Patterns box and click **Remove**.
- *Character Set printing widths* defines the printing width for each character in a character set. Change the printing widths in the Format Patterns box:
  - To add a printing width, click **Add...**. The New Character Width window is displayed. Enter the new width and click **OK**.

– To remove a printing width, select the printing width in the Format Patterns box and click **Remove**.

To set up your preferences for formatting, expand **Receipt Builder** and select **Formatters**. The formatting preferences are displayed. See Figure 2–3.

Figure 2–3 Preferences Window for Formatters



You can add and remove format patterns for each formattable type. Select the formattable type you want to change. The defined format patterns for that type are displayed in the Format Patterns box.

Note: These patterns need to match Java formatting. For more information, consult the Java API documentation for the Java class specified in parentheses next to the selected Formattable Types option.

- To add a new pattern, select **Add...**. The New Format Pattern window is displayed. Enter the new format and select **OK**.
- To remove a pattern, select the pattern in the Format Patterns box and click **Remove**.

Using the Receipt Data Object View

The Receipt Data Object view shows objects, classes, and methods defined in the `.ser` file. The Receipt Builder editor uses the RDO to access the methods provided from an object and the data contained within the object. You must attach the RDO to the blueprint to access the objects, classes, and methods.

In the Receipt Data Object view, you can perform the following functions:

- To attach the RDO for a blueprint, select the **Attach data object to receipt** button or **Attach RDO** in the View Menu. The Open RDO window is displayed. Select the `.ser` file you want to open and select **OK**.
- To detach the RDO shown in the Receipt Data Object view, select the **Detach data object from receipt** button or **Detach RDO** in the View Menu.
- If **Filter** is selected in the View Menu, methods that return `void` and methods with parameters (except for methods with the `Locale` parameter) are not included in the view since Receipt Builder cannot invoke these methods.
- If **Link** is selected in the View Menu, when you click a method element in the Graphical XML Editor view, the Outline and Receipt Data Object views are positioned to the place where the element is defined.

  For a serialized object that is attached through the Receipt Data Object view, only the methods for which there is any data are displayed. For example, if the attached `SaleReceipt.ser` file was produced by a sale that did not have a billing customer, the Receipt Builder tool is not able to display any methods when the method property is selected for the Billing Customer section in the `SaleReceipt.bpt` file.

  To access any missing methods in the Receipt Data Object view:

  1. Run the Point-of-Service client with the `persistBeansAsDataObject` value set to true. For more information, see "Creating a Serialized Business Object Using Point-of-Service".
  2. Perform a test case that includes the required data.
  3. Rename the `.ser` file produced in Step 2, for example, `SaleReceipt_wBillingCustomer.ser`.
  4. Attach the file to the Receipt Data Object view. The Receipt Builder tool drills down to the method when the corresponding method element is selected in the blueprint.
- If you right-click on any method or class in the Receipt Data Object view and select **Open Declaration**, the source code for that method or class is opened in the Graphical XML Editor view. The source code must be available in the project.

Best Practices for Using the RDO

Keep the following best practices in mind:

- Give preference to methods that have a `Locale` parameter over methods that do not. The locale enables the data to be printed on a receipt in the locale for the Point-of-Service customer's preferred language.
- If a method is defined for an object in both the class and interface branches, use the method from the interface branch of the tree.

Creating a Serialized Business Object Using Point-of-Service

To create a serialized business object file:

1. Ensure your `/pos/config/manager/BlueprintedDocumentManager.xml` has the `persistBeansAsDataObject` value set to true.

Caution: The value of true should only be used in a development environment since every transaction creates a serialized business object file when true is set. This could affect performance and security.

2. Run the Point-of-Service client and print the desired receipt. A new `.ser` file is added in the `/pos/receipts` directory.
3. If the `.ser` file is not already in the Eclipse workspace, add the serialized business object file to the Eclipse workspace for the Receipt Builder Tool. It is possible to drag and drop the file from Windows Explorer to the desired directory in the Eclipse Navigator or Package Explorer view. If the file already exists in the correct directory but does not show up, select the directory in the Eclipse Navigator or Package Explorer view, right-click, and select **Refresh**.

   Tip: When generating an RDO, include the information that pertains to the edits you are making to the receipt in the RDO. For example, to edit MasterCard information in a blueprint, perform a sales transaction (even in Training Mode) using a MasterCard as tender. Then, use the generated `SaleReceipt.ser` file as the RDO to attach when editing the receipt blueprint file.

Creating a Serialized Business Object Using Alternate Bean Creation

Instead of using Point-of-Service to create the serialized business object file, you can use the Object Inspector plug-in to create an instance of your receipt parameter bean if it has a zero-arg constructor. This may be more convenient than running Point-of-Service, but the resulting serialized object will not have any values in it to use when printing. Even so, you can use Object Inspector to execute simple set methods onto your bean instance. Be sure your class implements `PrintableDocumentParameterBeanIfc`.

To create the serialized file:

1. Select **File**, **New**, **Other...**, and then **Object**. The New window is displayed.
2. Expand **Java** and select **Serialized Object**. Click **Next**. The Create a Serialized Java Object window is displayed.
   a. Select the folder for the file.
   b. Enter the file name. Name your serialized business object file, `*.ser`, with the name that matches the name of the `*.bpt` file. For example, `NewReceipt.ser` and `NewReceipt.bpt`.
   c. Select the class that implements the `PrintableDocumentParameterBeanIfc`.
3. Click **Finish**. The serialized business object is created.

Editing a Receipt

The Receipt Builder tool recognizes `.bpt` file extensions as XML files that can be opened with the Graphical XML Editor.

Note: Eclipse displays a list of the available keyboard accelerators when you right-click on an element. You can select the function you want to perform from the list or use the keyboard as indicated.

To edit a receipt:

1. Open a Receipt Builder editor for the blueprint:
   a. Select **Navigate** and then **Open Resource...**. The Open Resource window is displayed. Double-click the desired blueprint file from the Navigator or Package Explorer view. You can also enter the name of the blueprint file.
   b. Click **Open**. A Receipt Builder editor for the file is opened in the Graphical XML Editor view.
2. For instructions on editing, see the following sections :
   - "Adding an Element"
   - "Changing the Number of Copies for a Receipt"
   - "Changing the Properties of an Element"
   - "Moving an Element"
   - "Removing an Element"
3. When you are done editing the receipt, close the Receipt Builder editor for the blueprint file. See "Closing a Receipt Editor".

Changing the Number of Copies for a Receipt

The number of copies to print for a receipt is a property of the receipt blueprint file. To change the number of copies:

1. Open the Outline view by clicking the Outline tab. If the view is not open, select **Window**, **Show View**, and then **Outline**.
2. Click the *Receipt Blueprint* element in the Outline view.
3. In the Properties view, select the General tab and change the Copies field.

Changing the Properties of an Element

The Properties view only displays the properties for the currently selected element. Properties can also be viewed by selecting the element in the Outline view.

There are two types of receipt elements:

- Static—Parts of the receipt that print as they appear in the Graphical XML Editor view. Headings are an example of a static element.

  You can edit static text directly:

  – In the Graphical XML Editor view, by clicking the text twice (slower than a double-click).

  – In the Properties view, by editing the Text field in the General tab.
- Dynamic—Parts of the receipt in which information is extracted from a business object and displayed on the printed receipt. The appearance of a dynamic element varies from what you see in the Graphical XML Editor view and what appears on the printed receipt.

Table 2–2 describes the properties that can be changed in the Properties view.

Table 2–2 Receipt Data Element Properties

Property	Receipt Component	Value	Description
ID	All	System-supplied.	Not printable. Shows the object identifier.
Method Stack	Method element	System-supplied.	Not printable. Shows the methods from the Receipt Data Object used to create the data.
Name	Report	User-supplied.	A unique name for the current report section in the blueprint.
Cut Paper	Report and Link Report	User-supplied.	If set to true, the paper is cut after the report completes. If set to false, the paper is not cut after the report completes. Defaults to true.
Depends On (Reports and Link Reports only)	Report and Link Report	User-supplied.	The method stack that this report depends on. If present, the method result must be non-null and true in order for the report to print. This value can only be set by selecting the method in the Receipt Data Object view and dragging and dropping it onto any type of report in the Outline view.
Document Type (for Link Report elements only)	Link Report	User-supplied.	The name of the blueprint type that should be printed at this location. The document type is matched to a blueprint through the configuration of the `BlueprintDocumentManager.xml` file.
Depends On Presence Of	Line, any element	User-supplied. Select a value from the list of available choices. Dependencies can include any method elements within the same receipt group.	Prints the line if the dependency assigned to it is present. A method element is considered not present if its output results in `null` or an empty string and its Is Printed When Length Zero value is set to false. Any element that depends on the presence of the non-present method element does not print.
Text	Element	User-supplied. Enter the value for this property.	Prints the text in the value field.
Repeat Next Line Also	Line	User-supplied. Select the check box to apply the Repeat Next Line Also property to the receipt line.	When checked, causes the next line to repeat with the current line: Line 1 Line 2 Line 1 Line 2 When not checked: Line 1 Line 1 Line 2 Line 2 This property is especially useful for subsequent repeating lines, for example, discount lines.

Table 2–2 (Cont.) Receipt Data Element Properties

Property	Receipt Component	Value	Description
Stretch	Any element	User-supplied. Select the check box to apply the Stretch property to the receipt data element.	Prints the element with as much space as it can without clipping other elements on the line. Multiple elements on the same line that are set to stretch are each weighted the same, that is, they grab equal amounts of white space.
Fill Char	Any element	User-supplied. Enter the value for this property.	Prints the element with the fill character up to the width of the element. The default fill character is a blank space.
Justified	Any element	User-supplied. Select a value from the list of available choices. The available values are left, center, and right.	Prints the selected receipt data element left-justified, centered, or right-justified within the space allotted to the element.
Escape(s)	Any element	User-supplied. Select the check box to apply an escape code to the receipt data element.	The selected receipt data element prints with a preceding escape code. ■ Escape codes must be prefixed with "\ |". ■ Escape codes can be found in the publicly available UnifiedPOS v1.9 documentation. ■ Multiple escape codes can be used on a single element. ■ Not every code is supported by each JPOS driver vendor. ■ Take care when using escape codes for paper cutting, scaling, reversing, and so on. ■ Escape codes have no affect on Console output.
Is Printed As Bar Code	Any element	User-supplied. Select the check box to apply the Is Printed As Barcode property to the receipt data element.	Prints the selected receipt data element as a barcode.
Is Preceded By Space	Method element	User-supplied. Select the check box to apply the Is Preceded By Space property to the receipt data element.	Prints a space before the selected element.
Is Printed When Length Zero	Method element	User-supplied. Select the check box to apply the Is Printed When Length Zero property to the receipt data element.	If checked, method elements that produce no printable output are still considered printed. Otherwise, unchecked nulls and empty strings are considered not printed. Any additional elements that depend on this element do not print.
Is Printed When Value Zero	Method element that returns a number	User-supplied. Select the check box to apply the Is Printed When Value Zero property to the receipt data element.	If checked, prints the value returned as zero. If unchecked, does not print values returned as zero. Useful for static elements that depend on non-zero values.

Table 2–2 (Cont.) Receipt Data Element Properties

Property	Receipt Component	Value	Description
Fixed Width	Method element	User-supplied for dynamic elements. Enter the fixed number of character spaces.	Prints only the fixed number of characters for a dynamic element. If the fixed width is set longer than the output produced, the fill character for the element is used. For example, if you set the length of a customer first name object to five, only the first five letters of the customer's first name are printed.
Format Mask	Method element	User-supplied. Enter the value.	Formats the output by the selected format pattern. The format patterns available depend on the data type of the element, that is, Date, String, Number (including Currency), or Boolean.
Value Printed When True	Method element that returns a Boolean	User-supplied. Enter the value.	Prints this value when the method element produces a Boolean value equal to true. Defaults to true.
Value Printed When False	Method element that returns a Boolean	User-supplied. Enter the value.	Prints this value when the method element produces a Boolean value equal to false. Defaults to false.
Prefix	Method element	User-supplied. Enter the value.	Prints the prefix in front of the output for the element. Prints regardless of whether the element produced any output.
Suffix	Method element	User-supplied. Enter the value.	Prints the suffix immediately after the output for the element. Prints regardless of whether the element produced any output.
File Name	Image element	User-supplied. Enter the file name.	Prints an image. Refers to any image file in the `bin/` directory.

Adding an Element

A flyout palette is available on the right in the Graphical XML Editor view. It enables you to add the blueprint elements described in Table 2–3.

Table 2–3 Elements that Can Be Added to a Blueprint

Element	Description
Report	Adds a new report to the blueprint. When this element is selected, the New Report window is displayed. Enter the new report name, select if the paper is cut at the end of the report, and click **OK**.
Link Report	Adds a link to another blueprint. When this element is selected, the New Link window is displayed. Enter the name of the blueprint file to be linked to the receipt and click **OK**.
Group	Adds a group.
Line	Adds a line.
Element	Adds a static text element.

Table 2–3 (Cont.) Elements that Can Be Added to a Blueprint

Element	Description
Copy Element	Adds an element with copy sensitive text, for example: ■ Customer Copy ■ Merchandise Copy ■ Store Copy
Image	Adds a static image. When this element is selected, the Open Image window is displayed. Select the image to be added to the receipt and click **OK**.
Date	Adds an element that prints the current date and time, depending on the format mask applied, at the time of printing.

To add an element to a receipt:

1. Open the flyout palette in the Graphical XML Editor view by clicking the Show Palette button on the right.
2. Click the element in the palette you want to add.
3. Click the place in the receipt where you want to add the element.

   The editor only allows you to add an element in a valid place in the receipt. For example, it prevents you from adding a group within a group. A black bar appears at valid drop locations.

Note: You must add receipt groups and lines before you can add receipt elements:

- Reports can only be added to the root blueprint element.
- Groups can only be added to reports.
- Lines can only be added to groups.
- Elements, any type, can only be added to lines.

Moving an Element

The easiest way to move an element is to drag and drop the element. You can also select the element and then use the move arrow buttons in the toolbar, context menu, or Edit menu, to move the element.

Lines, groups, and reports can only be moved up and down. A static or dynamic element can only be moved from one group to another by drag and drop or cut and paste.

Note: If drag and drop does not result in moving the element to the desired location, try Undo (**CTRL-Z**) and cut and paste instead. An element can only be pasted if the target line or element is selected.

Removing an Element

Select the element you want to remove in the Graphical XML Editor or Outline view. Select **Delete** from the Edit or context menu.

Saving a Receipt

To save any changes made to the receipt currently being edited, select **File** and then **Save**. The updated receipt is saved.

Closing a Receipt Editor

To close an editor, select **File** and then **Close**. If there any unsaved changes, the Save Resource window is displayed:

- To save the changes and close the editor, select **Yes**.
- To close without saving the changes, select **No**.
- To cancel closing the editor, select **Cancel**.

Printing a Receipt

To print the receipt, select **File** and then **Print**. The receipt is displayed in the Console view. It does not print to a printer. If a serialized object is attached in the Receipt Data Object view, the receipt is displayed to the console with the data from the object. The logo and any barcodes do not display in the Console view.

To print the actual blueprint XML, select the blueprint file from the Navigator or Package Explorer view, right-click, and choose **Open With, XML Editor**. If **XML Editor** is not an option, choose **Other...**, **XML Editor**, and then **OK**. From this editor, the Print menu option prints to the system's configured printer.

Note: Sardine and Eclipse support multibyte output.

Printing to a JPOS Printer

Printing the receipt output for an edited blueprint to an actual JPOS printer can be accomplished from the Point-of-Service client installation. The `TestReceiptPrinter` class can run the Dispatcher and print to the POS printer configured in the `posdevices.xml` file in the Point-of-Service client installation.

To print to a JPOS printer:

1. Set the classpath, and configure the `jpos.xml` and `posdevices.xml` files.
2. From the `bin/` directory, run the following.

```
java oracle.retail.stores.pos.receipt.blueprint.TestReceiptBlueprint <bpt path>
<ser path>
```

Exiting the Receipt Builder Tool

To exit the Receipt Builder Tool, exit Eclipse by selecting **File** and then **Exit**. If there are any unsaved changes for any open receipts, the Save Resource window is displayed:

- To save the changes and exit, select **Yes**.
- To exit without saving the changes, select **No**.
- To cancel exiting the Eclipse, select **Cancel**.

Creating a New Receipt

To enable Point-of-Service to print new output:

1. Create a serialized object of the new data to be printed. See "Creating a Serialized Business Object Using Point-of-Service" and "Creating a Serialized Business Object Using Alternate Bean Creation".
2. Add a <RECEIPT> entry to the `/pos/config/manager/BlueprintedDocumentManager.xml` file. The entry maps the name of the new receipt to the blueprint file that will be created for it.

   ```
   <RECEIPT type="MyNewReceipt" fileName="MyNewReceipt.bpt" />
   ```

   In these steps, `MyNewReceipt` is used as the name of the receipt being added.
3. Add Java code to Point-of-Service, in a Site or Aisle for example, to print the receipt. The following is an example of the code to add:

   ```
   PrintableDocumentManagerIfc pdm =
   (PrintableDocumentManagerIfc)bus.getManager(PrintableDocumentManagerIfc.TYPE);
   ReceiptParameterBeanIfc bean = getMyParameterBean();
   bean.setDocumentType("MyNewReceipt");
   pdm.printReceipt((SessionBusIfc)bus, bean);
   ```

4. To create the RDO, run the Point-of-Service client with the new code. The client may fail if the missing blueprint exception is not caught.
5. Create the blueprint file. See "Creating a New Blueprint File".
6. Attach `MyNewReceipt.ser`, created in Step 4, to the Receipt Data Object view. See "Using the Receipt Data Object View".
7. Format the content. See "Editing a Receipt".

Creating a New Blueprint File

To create a new blueprint file:

1. Select **File**, **New**, and **Other...**. The New window is displayed.
2. Expand **Oracle-Retail** and select **Receipt Blueprint**.
3. Click **Next**. The Create a Receipt Blueprint XML File window is displayed.
4. Select the parent folder and enter the file name.
5. Click **Finish**. The blueprint file is created and an editor is opened for the file.
6. Attach the serialized business object file to the blueprint file:
   a. In the Receipt Data Object view, select the **Attach data object to receipt** button. The Open RDO Ending In "*.ser" window is displayed.
   b. Select the serialized business object file.
   c. Click **OK**. The serialized business object file is attached to the blueprint file.
7. For information on editing the new blueprint file, see "Editing a Receipt".

Tips for Using XML Receipts and the Receipt Builder Tool

Keep the following tips in mind:

- Create a custom Plain Old Java Object (POJO), if needed, to encapsulate complicated output.
- It may be necessary to add a new method to an object for receipt printing purposes, such as, `TaxInformationContainer.getReceiptCode()`.

- Reattaching a changed Receipt Data Object (RDO) class reloads the classes and methods in the Receipt Data Object view in the Eclipse window.
- For additional logging, add the DEBUG logger category for `oracle.retail.stores.pos.receipt.blueprint` to the `log4j.xml` file.
- Consider JMS-based payload sizing restrictions for file transfer.
- If a receipt line gets truncated due to both the width of the receipt paper and the content that is expected on the receipt line, you may want to break the specific line into two lines so that the information is not truncated.

A

Appendix: Blueprint Files

Table A–1 has a description of each of the available blueprint files. To see the blueprint files specifically for a fiscal printer, see Table A–2.

Table A–1 Description of Available Blueprint Files

Blueprint Name	Description
AlterationReceipt.bpt	When a sale transaction includes alteration items and instructions, the instructions are printed separately on this receipt.
Ankle.bpt	The ankle is printed above the footer in a receipt.
AssociateProductivityReport.bpt	This report summarizes the amount of nontaxable and taxable sales for a sales associate in a given date range.
AvailableToPromiseInventorySlip.bpt	This slip is printed for a pickup order.
BillPayFooter.bpt	This footer is printed at the bottom of the bill pay transaction receipt before the receipt footer information.
BillPayHeader.bpt	This header is printed below the logo on a bill pay transaction receipt.
BillPaymentReport.bpt	This report summarizes the bill pay transactions in a given date range.
BillPayReceipt.bpt	This receipt is printed for a bill payment transaction.
CanceledReceipt.bpt	This receipt is printed for a cancelled transaction.
CashRounding.bpt	This section is printed if there is a cash rounding adjustment made for a transaction.
CreditCardPromotionDisclosure.bpt	This slip is printed for a credit card promotion.
CreditSignatureSlipReceipt.bpt	This signature slip is printed for a transaction tendered with a credit card.
CustomerSendInfo.bpt	This section is used to print customer information when items are scheduled for pickup or delivery.
DebitSlipReceipt.bpt	This receipt is printed for a transaction tendered with a debit card.
DepartmentSalesReport.bpt	This report summarizes the amount of nontaxable and taxable sales for each department in a given date range.
DepositedCheck.bpt	This section is printed if a check is used as tender for a transaction.
ECheck.bpt	This section is printed if an eCheck is used as tender for a transaction.

Table A–1 (Cont.) Description of Available Blueprint Files

Blueprint Name	Description
ECheckAuthorizationApproval.bpt	This section is printed if the authorization text for approval of an eCheck is needed.
ECheckSignatureSlipReceipt.bpt	This slip is printed if a transaction tendered with an eCheck.
EmployeeDiscountReceipt.bpt	This receipt is printed for a sale transaction that includes an employee discount.
ExchangeReceipt.bpt	This receipt is printed for an exchange transaction.
ExternalOrder.bpt	This section is printed if the transaction includes an external order.
Footer.bpt	The footer is printed at the bottom of the receipt after all transaction information and the ankle.
Franking.bpt	This section is printed if any of the tenders were franked.
FrankingHeader.bpt	This header is printed at the top of the section for any tenders that were franked.
GiftCardInquirySlip.bpt	This slip is printed for a gift card inquiry.
GiftCertificate.bpt	This section is printed if a gift certificate is issued.
GiftCertificateTendered.bpt	This section is printed if a gift certificate is used as tender for a transaction.
GiftReceipt.bpt	This receipt is printed when a gift receipt is needed for a sale transaction.
Header.bpt	The transaction header is printed below the logo in a receipt.
HourlyProductivityReport.bpt	This report summarizes sales, returns, net sales, and percent of net total by the hour in a given date range.
HousePaymentReceipt.bpt	This receipt is printed when a house account payment is made.
ICCDetails.bpt	This receipt is printed when an Integrated Chip Card is used for a transaction.
InstantCredit.bpt	This section is printed if instant credit is used as tender for a transaction.
InstantCreditInquiryInfoSlip.bpt	This slip is printed for a house account inquiry.
InventoryInquirySlip.bpt	This slip is printed for an inventory inquiry.
Items.bpt	This section of a receipt lists the items in the transaction.
LayawayDeleteReceipt.bpt	This receipt is printed when a layaway is deleted.
LayawayPaymentReceipt.bpt	This receipt is printed when a payment is made on a layaway transaction.
LayawayPickupReceipt.bpt	This receipt is printed when the items in a layaway are picked up.
LayawayReceipt.bpt	This receipt is printed for a layaway transaction.
Logo.bpt	This section is printed at the top of a receipt.
MailBankCheck.bpt	This section is printed if a refund is issued as a check.
MailBankCheckSlip.bpt	This slip is printed if a refund is issued as a check.
MallCertCheck.bpt	This section is printed if a mall gift certificate is used as a check tender for a transaction.
MallCertPO.bpt	This section is printed if a mall gift certificate is used as a purchase order tender for a transaction.

Table A–1 (Cont.) Description of Available Blueprint Files

Blueprint Name	Description
MallGiftCert.bpt	This section is printed if a mall gift certificate is used as tender for a transaction.
MoneyOrder.bpt	This section is printed if a money order is used as tender for a transaction.
NameVerificationSlip.bpt	This slip is printed when verification of the name of the card holder is required.
NoSaleSlip.bpt	This slip is printed for a no sale transaction.
OrderPrintHeader.bpt	This header is printed at the top of an order receipt.
OrderReceipt.bpt	This receipt is printed for an order transaction.
OrderSummaryReport.bpt	The report summarizes the number of orders by status and the amount of each order type in a given date range.
OrderStatusReport.bpt	This report shows information for new, printed, partial, filled, cancelled, and completed orders in a given date range.
PAT.bpt	This content is printed for the footer in a receipt when customer information is collected for a Patriot Act (PAT) cash transaction.
PickupDeliveryOrderReceipt.bpt	This receipt is printed if a transaction contains an item that has been designated for pickup or delivery.
PurchaseOrder.bpt	This section is printed if a purchase order is used as tender for a transaction.
QueuedTransactionsReport.bpt	This report shows information about queued transactions.
RebateReceipt.bpt	This receipt is printed when a customer is eligible for a rebate.
RedeemReceipt.bpt	This receipt is printed when a gift card, gift certificate, or store credit is redeemed.
ReturnDeniedReceipt.bpt	This receipt is printed when items being returned are denied for return by Oracle Retail Returns Management.
ReturnReceipt.bpt	This receipt is printed for a return without a receipt.
SaleReceipt.bpt	This receipt is printed for a sale transaction.
ShippingSlip.bpt	This slip is printed when shipping information is included in a transaction.
SpecialOrderCancelReceipt.bpt	This receipt is printed when an order is cancelled.
SpecialOrderCompleteReceipt.bpt	This receipt is printed when a customer pays the remaining balance on an order and picks up the merchandise.
SpecialOrderReceipt.bpt	This receipt is printed for an order transaction.
StoreCredit.bpt	This section is printed if store credit is used as tender for a transaction.
StoreCreditReceipt.bpt	This receipt is printed when a store credit is issued.
SummaryReport.bpt	This report shows tender, transaction, and statistical information for a store, register, or till.
Survey.bpt	This customer survey/reward is printed at the end of a receipt.
SuspendedTransactionsReport.bpt	This report shows information about suspended transactions.
SuspendReceipt.bpt	This receipt is printed when a transaction is suspended.
TaxExemptReceipt.bpt	This receipt is printed when a sale transaction includes tax exempt items.

Table A–1 (Cont.) Description of Available Blueprint Files

Blueprint Name	Description
TempShoppingPass.bpt	When enrollment for a house account is approved, this temporary shopping pass is printed.
Tenders.bpt	This section of a receipt lists the tenders used in the transaction.
TillCountReport.bpt	This report shows the results of a till count.
TillLoanReceipt.bpt	This receipt is printed for a till loan transaction.
TillPayInReceipt.bpt	This receipt is printed for a till payin transaction.
TillPayOutReceipt.bpt	This receipt is printed for a till payout transaction.
TillPayrollPayOutReceipt.bpt	This receipt is printed for a till payroll payout transaction.
TillPickupReceipt.bpt	This receipt is printed for a till pickup transaction.
TillSummaryReport.bpt	This report shows tender, transaction, and statistical information for a till.
Totals.bpt	This section of a receipt lists the tender totals for the transaction.
TravellersCheck.bpt	This section is printed if traveller's checks are used as tender for a transaction.
VATSummary.bpt	This section of a receipt lists the Value Added Tax (VAT) information for the transaction.
VoidBillPayReceipt.bpt	This receipt is printed when a bill payment transaction is post-voided.
VoidCashRounding.bpt	This section is printed if a cash rounding adjustment was made for a transaction that is post-voided.
VoidDepositedCheck.bpt	This section is printed if a check was used as tender for a transaction that is post-voided.
VoidHousePaymentReceipt.bpt	This receipt is printed when a house payment transaction is post-voided.
VoidLayawayReceipt.bpt	This receipt is printed when a layaway transaction is post-voided.
VoidRedeemReceipt.bpt	This receipt is printed when a redeem transaction is post-voided.
VoidSaleReceipt.bpt	This receipt is printed when a sale transaction is post-voided.
VoidSpecialOrderReceipt.bpt	This receipt is printed when an order transaction is post-voided.
VoidStoreCredit.bpt	This section is printed if a store credit was used as tender for a transaction that is post-voided.
VoidStoreCreditIssued.bpt	This section is printed if a store credit was issued as a part of a transaction that is post-voided.
VoidTillAdjustmentsReceipt.bpt	This receipt is printed when a transaction for any till adjustment is post-voided.

Table A–2 has a description of each of the available blueprint files for a fiscal printer.

Table A–2 Description of Available Blueprint Files for a Fiscal Printer

Blueprint Name	Description
FiscalAnkle.bpt	The ankle is printed above the footer in a receipt.
FiscalBarcode.bpt	This section shows the barcode for the transaction.
FiscalEmployeeDiscountReceipt.bpt	This receipt is printed for a sale transaction that includes an employee discount.
FiscalEmployeeDiscountReceiptMessages.bpt	This section shows messages for an employee discount.
FiscalFooter.bpt	The footer is printed at the bottom of the receipt after all transaction information and the ankle.
FiscalHeader.bpt	The transaction header is printed below the logo in a receipt.
FiscalItems.bpt	This section of a receipt lists the items in the transaction.
FiscalLayawayDeleteReceipt.bpt	This receipt is printed when a layaway is deleted.
FiscalLayawayDeleteReceiptMessages.bpt	This section shows messages for a layaway delete.
FiscalLayawayPickupReceipt.bpt	This receipt is printed when the items in a layaway are picked up.
FiscalLayawayPickupReceiptMessages.bpt	This section shows messages for a layaway pickup.
FiscalLayawayReceipt.bpt	This receipt is printed for a layaway transaction.
FiscalLayawayReceiptMessages.bpt	This section shows messages for a layaway.
FiscalPDOReceiptMessages.bpt	This section shows messages for a pickup or deliver order.
FiscalPickupDeliveryOrderReceipt.bpt	This receipt is printed if a transaction contains an item that has been designated for pickup or delivery.
FiscalReturnReceipt.bpt	This receipt is printed for a return without a receipt.
FiscalSaleReceipt.bpt	This receipt is printed for a sale transaction.
FiscalSaleReturnReceiptMessages.bpt	This section shows messages for a return.
FiscalSpecialOrderReceipt.bpt	This receipt is printed for an order transaction.
FiscalSpecialOrderReceiptMessages.bpt	This section shows messages for a special order.
FiscalTaxExemptReceipt.bpt	This receipt is printed when a sale transaction includes tax exempt items.
FiscalTotals.bpt	This section of a receipt lists the tender totals for the transaction.

Glossary

blueprint file

Set of instructions used to print a receipt or Point-of-Service report to a POS printer. The instructions are serialized to disk as XML files with a `.bpt` extension. The `.bpt` file extension is used to enable the Eclipse IDE to assign the specific editor from the Receipt Builder Tool plug-in for blueprint files.

Eclipse

IDE used by developers and implementers.

element (receipt)

A member of a blueprint file that produces printable output:

- Static elements are printed exactly as they appear in the blueprint file.
- Dynamic elements, like method-based elements, have text created by executing a stack of methods against the RDO. Examples include dates, image, and copy elements.

element (XML)

Any angle-bracket section of XML text, that is, `<blueprint>...</blueprint>`. Potential elements, as specified by the XSD for receipts, include the following:

- blueprint
- report
- linkReport
- group
- line
- element
- copyElement
- imageElement
- methodElement
- dateTimeElement
- method

Graphical Editing Framework (GEF)

An API provided by Eclipse to enable plug-ins to be created with simple WYSIWYG editors. This API is used by the Receipt Builder Tool.

group

Collection of printable elements. Lines are grouped for the purpose of iteration. If a group contains a method-from-a-method element that is part of a list or an array, the entire group is repeated for each iteration of the array.

line

A single line in a receipt. It may contain multiple printable elements. If a line contains a method-from-a-method element that is part of a nested list or array, the entire line is repeated for that nested array.

plug-in

Application-specific Java code modules that, when added to the configuration of an existing application, add new functionality to that application. Code, XML files, and resources, such as images, can be part of a plug-in.

POS printer

Refers to a simple printer used by the Point-of-Service client to print sale receipts. The font is always fixed-width and the printer typically supports 38 to 44 characters in a single line depending on the font size. The JPOS API enables Java programmers to produce code that prints to a POS printer. Each brand of printer has it own JPOS driver.

Receipt Data Object (RDO)

The dynamic data being printed. The data object is a Java object that exists in memory at Point-of-Service client runtime. The data object can be saved to disk as a `.ser` file by the Point-of-Service client using default Java serialization, if this option is turned on. Point-of-Service code refers to each of these objects as a `PrintableDocumentParameterBean`.

report

Collection of groups within a blueprint file. The receipt paper is typically cut after a report. Reports can be printed multiple times for copies.

What You See Is What You Get (WSYIWYG)

Term for applications where what is displayed on the screen is what is produced when printed.

Oracle Corporation
World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065
U.S.A.

Worldwide Inquiries:
Phone: +1.650.506.7000
Fax: +1.650.506.7200
oracle.com

Primary Author: Bernadette Goodman