

PeopleSoft XML Publisher

Primer

This document is a Primer on using XML Publisher for PeopleSoft. It is written by David Vandiver.

Table of Contents

Contents

- Install 2
 - Download Word Plugin 2
 - Verify Install..... 3
 - XDO Config File 4
- Create the Template 6
 - Create a basic Query..... 6
 - Capture the XML Output..... 7
 - Building Our First RTF 9
 - Reviewing the Output in PDF 10
- Connecting PeopleSoft to the Template 11
 - Creating an XML Data Source 11
 - Creating the Report 12
 - Building a Page 15
 - Migration Concerns 17
- Advanced Template Building..... 19
 - Tables 19
 - IF Conditioning 19
 - Headings and Footers 19
 - Bar Codes 19

Install

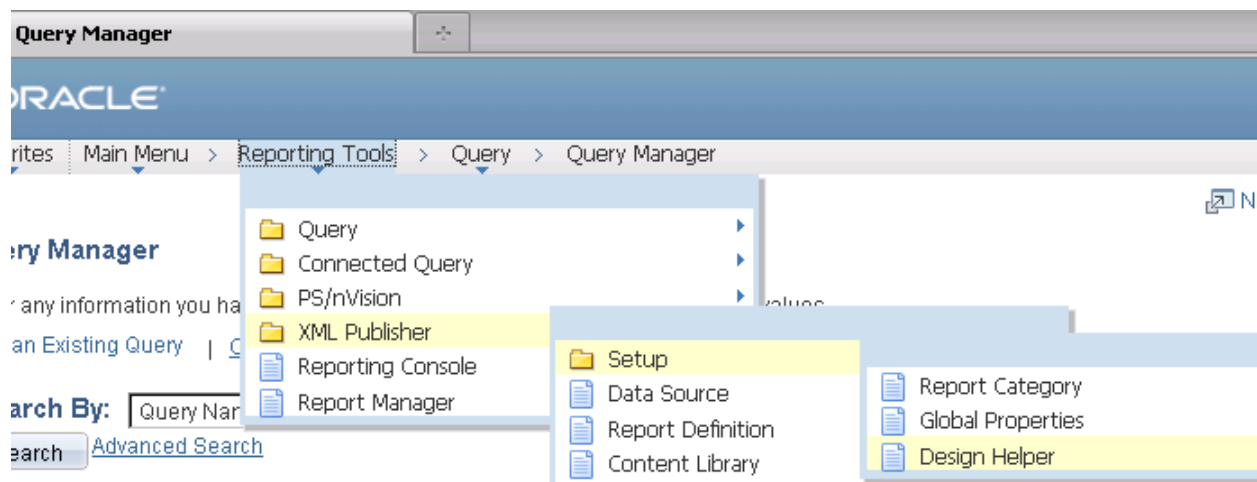
Welcome to our first session on XML Publisher for PeopleSoft. In this session, we will discuss how to install the plugin, verify the install, and create the xdo configuration file. This training series will focus on the development aspect and assumes that XML Publisher is already configured on the application server by the system administrator.

Download Word Plugin

The first thing we need to do is download the Word plugin, which is stored within the PeopleSoft application. You can go to any of your PeopleSoft environments, including Demo.

Once in PeopleSoft, navigate to

- Reporting Tools
- XML Publisher
- Setup
- Design Helper



This will open a page with a single link. Click the url link to download the Plug-In for Microsoft Word. In this session, we are using Firefox, so the download might look differently from other browsers. We will save the file to the default location Downloads. Once the download is complete, we will open the folder to find the zip file. We now must unzip the file. We will extract the zip file contents in the same folder.

Once the file is unzipped, open the XMLP_DESKTOP folder and locate the BI Publisher Desktop file. Open this file to begin the Word Plug In installation.

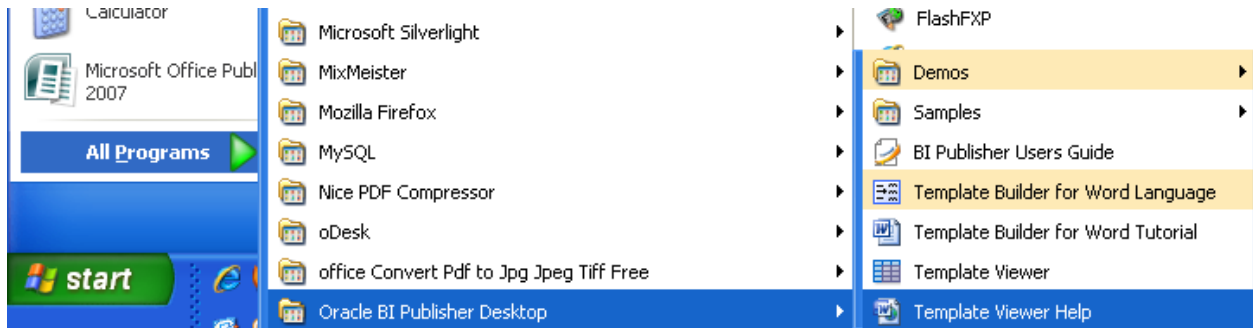
We will accept the defaults during the installation process. This install not only gives us the Word Plug In, but also installs a Start Menu folder with demos, samples, PDF User Guide, and other development tools.

When the install is done, click Finish.

Verify Install

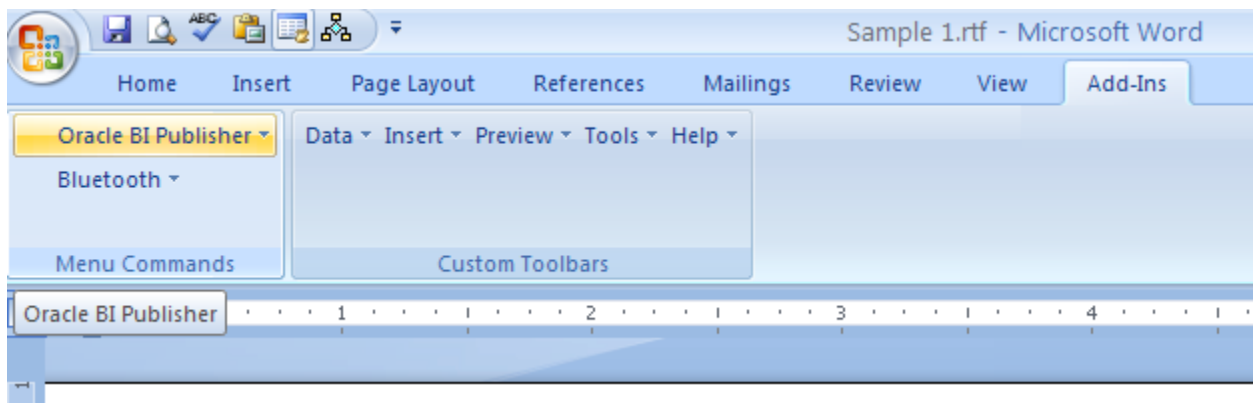
There are two methods to check the installation.

The first method is to review the Start Menu folder called Oracle BI Publisher Desktop. In this folder, there should be a Demos and Samples folder, as well as a User Guide and other development tools. The user Guide will come in handy as you advance in your development skills of BI Publisher.



The second method is to open Microsoft Word and review the Add In.

If you have Word 2007 or 2010, your experience should look similar to what appears in this training video. You should have an Add Ins menu option on the far right. Once you go to this menu, you should see the Oracle BI Publisher Ribbon.



There are two ways to navigate to most of the actions. You can click the Oracle BI Publisher menu options. You will notice that the items under Insert are the same as the Insert menu, but sometimes in a different order.

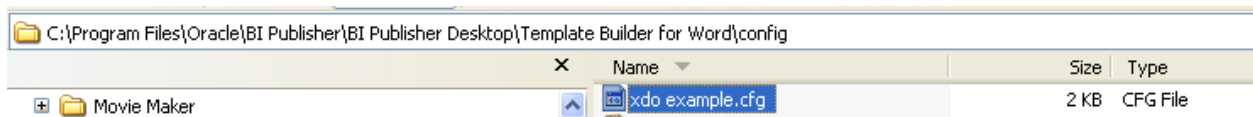
One note to point out. For Peoplesoft installations using Oracle BI Publisher, the Log On menu option is not used.

As your preview the menu options, you will see there are actions to load the XML Data and Schema files, insert fields and conditional formatting, preview in an assortment of formats, and additional developer tools to assist in creating XML Publisher reports.

XDO Config File

One final item you may need to configure on the desktop installation is the xdo file. Both the desktop and server can and should have an xdo configuration file. The desktop xdo file is only needed for development, and can be configured as needed. By default, an xdo.cfg file is not installed on the desktop.

If you plan to develop reports needing MICR or Bar code fonts, you will need to configure the desktop xdo file. Navigate to the Program Files directory, and locate Oracle, BI Publisher, BI Publisher Desktop, Template Builder for Word, Config. You can copy the xdo.sample.cfg file and name the new file xdo.cfg. If you cannot see your file extensions, then just name the file xdo and ensure the Type remains as a CFG File.



Open the xdo file in an editor and review the fonts section on examples. We will cover fonts, such as MICR and Bar Codes, in a later session.

```
xdo example.cfg X
0 10 20 30 40 50 60 70
1 <config version="1.0.0" xmlns="http://xmlns.oracle.com/oxp/config/">
2   <!-- Properties -->
3   <properties>
4     <!-- System level properties -->
5
6     <!-- PLEASE SELECT A VALID TEMPFILE DIRECTORY!!! -->
7     <property name="system-temp-dir">c:/Temp</property>
8
9     <!-- PDF compression -->
10    <property name="pdf-compression">>true</property>
11
12    <!-- PDF security -->
13    <property name="pdf-security">>false</property>
14    <property name="pdf-open-password">user</property>
15    <property name="pdf-permissions-password">owner</property>
16    <property name="pdf-no-printing">>true</property>
17    <property name="pdf-no-changing-the-document">>true</property>
18  </properties>
19
20
21  <!-- Font setting -->
22  <font>
23
24    <!-- Windows Wingdings font -->
25    <font family="Wingdings" style="normal" weight="normal">
26      <truetype path="C:\WINNT\fonts\wingding.ttf" />
27    </font>
28
29    <font family="GnuMICR" style="normal" weight="normal">
```

As a final reminder, this xdo file only applies for local development on this desktop. The server's xdo file will also need to be modified if adding new fonts to a report.

You now have the Oracle BI Publisher installed on your machine. In our next session, we will create a query and start development in Word.

Create the Template

Create a basic Query



We are going to create a basic query that joins the VOUCHER and DISTRIB_LINE tables.

Query Name: XML_TRAINING_VCHR

Click folder next to record to show fields. Check fields to add additional records by clicking the records tab. When finished

Chosen Records

Alias Record

-  A VOUCHER - AP Voucher Header Table
-  B DISTRIB_LINE - Voucher Distribution Table

We'll show some basic fields, like Voucher ID, Invoice Date, Account, Department, etc.

Fields							Customize Find View All
Col	Record.Fieldname	Format	Ord	XLAT	Agg	Heading Text	
1	A.BUSINESS_UNIT - Business Unit	Char5				Unit	
2	A.VOUCHER_ID - Voucher ID	Char8				Voucher	
3	A.INVOICE_ID - Invoice Number	Char30				Invoice	
4	A.INVOICE_DT - Invoice Date	Date				Date	
5	A.VENDOR_SETID - Vendor SetID	Char5				Vndr SetID	
6	A.VENDOR_ID - Vendor ID	Char10				Vendor	
7	B.VOUCHER_LINE_NUM - Voucher Line Number	Num5.0				Line	
8	B.DISTRIB_LINE_NUM - Distribution Line	Num5.0				Distribution Li	
9	B.DESCR - Description	Char30				Descr	
10	B.ACCOUNT - Account	Char10				Account	
11	B.DEPTID - Department	Char10				Dept	
12	B.MONETARY_AMOUNT - Monetary Amount	SNm25.3				Amount	

We will make the query be Public.

Query Properties

*Query:

Description:

Folder:

*Query Type:

*Owner:

Distinct Security Join Optimizer

Query Definition:

We will make the criteria be based on two prompts (Business Unit and Voucher ID)

Criteria			
Logical	Expression1	Condition Type	Expression 2
<input type="text" value=""/>	A.BUSINESS_UNIT - Business Unit	equal to	B.BUSINESS_UNIT - Business Unit
AND	A.VOUCHER_ID - Voucher ID	equal to	B.VOUCHER_ID - Voucher ID
AND	A.BUSINESS_UNIT - Business Unit	equal to	:1
AND	A.VOUCHER_ID - Voucher ID	equal to	:2

Lastly, we will run the query to ensure the query functions. It is best to have as many of the fields in the query, before working on the Word RTF Template. Add any fields that you might remotely need, as adding them to the query later adds extra effort.

Capture the XML Output

To run this next piece, we need to save the query and go back to the Query Manager window.

Search for your query again, and choose XML.

Query									
Select	Query Name	Descr	Owner	Folder	Edit	Run to HTML	Run to Excel	Run to XML	Schedule
<input type="checkbox"/>	XML_TRAINING_VCHR		Public		Edit	HTML	Excel	XML	Schedule

You should receive a prompt page (if you have criteria prompts). Enter sample values and then press enter. Depending on your browser, save the file. You may want to save the file with a portion of your sample data. These XML files will be referred to as XML Data files.

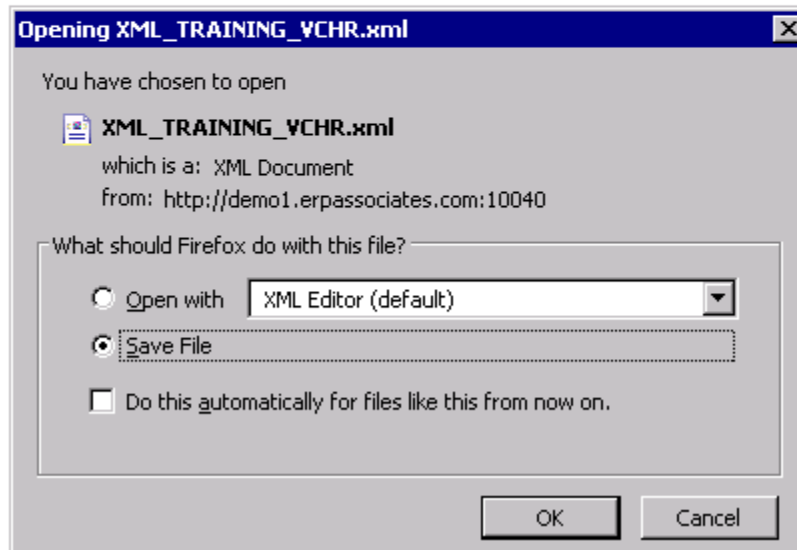
XML_TRAINING_VCHR

Unit:

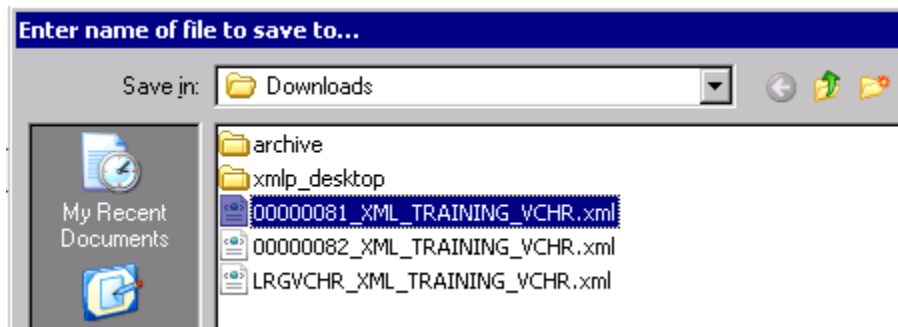
Voucher:

[View Results](#)

Unit	Voucher	Invoice	Date	Vndr SetID	Vendor	Line	Distributio
------	---------	---------	------	------------	--------	------	-------------

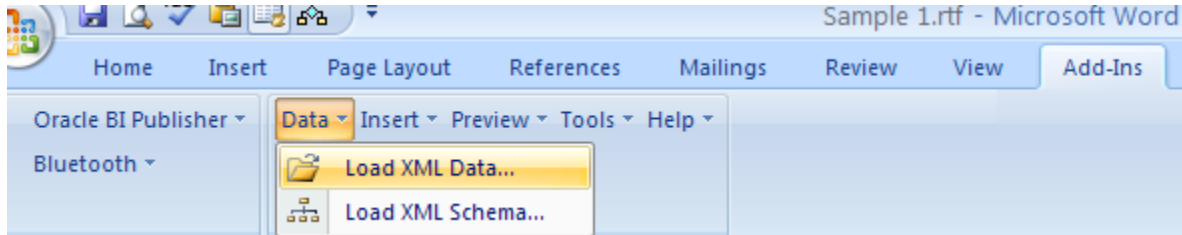


If you have to add more fields to your query after building the Word RTF template, you should probably recreate the XML Data files.



Building Our First RTF

Open Microsoft Word, and go to the Add In tab. Click Data and choose Load XML Data.



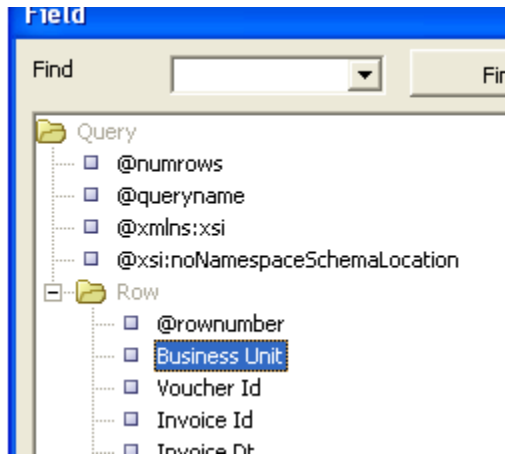
There is an option to load the Data and Schema separately, but with the XML file that was created with the Query tool, it has both the schema and data. So we can just load using the "Load XML Data".

Note: If you come back to the RTF file later, you need to reload the XML Data each time.

You add fields by choosing Insert → Field. You can drag and drop the fields

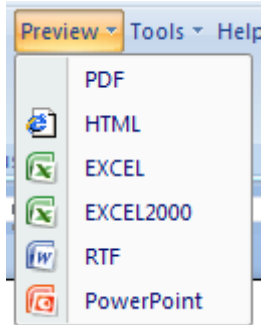
BUSINESS_UNIT

VOUCHER_ID



Reviewing the Output in PDF

Choose Preview → PDF. This should open the report in your Adobe Reader.

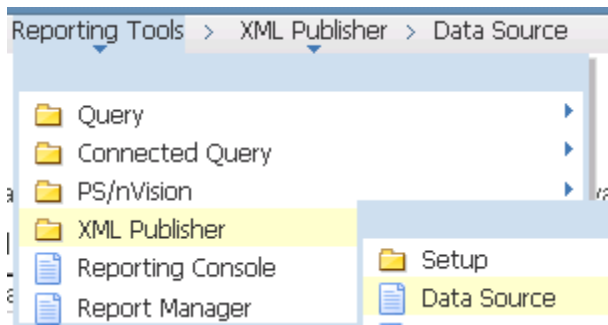


There is a Preview → RTF, but most of your reports will probably be PDF output. Though a bit confusing, you will build the template in RTF, but have the output be in PDF.

Connecting PeopleSoft to the Template

Creating an XML Data Source

Navigate to Reporting Tools → XML Publisher → Data Source → Add



Pick PS Query for your Data Source Type, then pick your query name for the Data Source ID.

Data Source

[Find an Existing Value](#)

| [Add a New Value](#)

Data Source Type:

PS Query

***Data Source ID:**

XML_TRAINING_VCHR



Private Data Source

Add

[Find an Existing Value](#)

| [Add a New Value](#)

After you click ADD, Enter a description and the Object Owner ID, then click both GENERATE links for the Sample Data File and Schema File. Then press SAVE.

Data Source

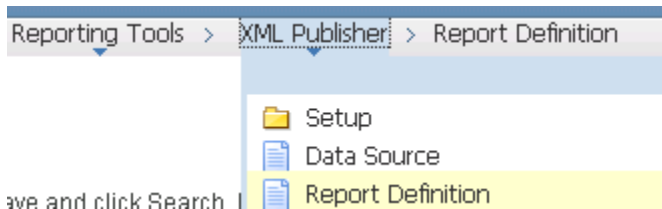
Data Source Type: PS Query
Data Source ID: XML_TRAINING_VCHR

Data Source Properties				
Description:	<input type="text" value="XML Training Voucher"/>		<input checked="" type="checkbox"/>	Active
Object Owner ID:	<input type="text" value="Payables"/>			
Registered Date/Time:	06/06/12 12:27:07AM	Registered By:	VP1	
Last Update Date/Time:		Updated By:		

Related Files				
File Type	File	Generate File	Or	Upload File
Sample Data File	XML_TRAINING_VCHR.XML	Regenerate	Or	Upload
Schema File	XML_TRAINING_VCHR.XSD	Regenerate		

Creating the Report

Navigate to Reporting Tools → XML Publisher → Report Definition → Add



Enter a Report Name. Use the Data Source Type PS Query, and enter the Query name. Then select ADD.

Report Definition

[Find an Existing Value](#) | [Add a New Value](#)

Report Name:

***Data Source Type:**

Data Source ID:

On the first page of the Report Definition, enter the following:

- Report Description
- Change the Status to Active
- Pick a Report Category. ALLUSER most likely is the best default if you do not utilize Category ID Security.
- Leave the Template Type as RTF

Definition	Template	Output	Properties	Security	Bursting
Report Name:	XML_VCHR				
Data Source					
Data Source Type:	PS Query				
Data Source ID:	XML_TRAINING_VCHR				
Data Source Description:	XML Training Voucher				
Report Properties					
Report Description:	<input type="text" value="XML Voucher"/>				
*Report Status:	<input type="text" value="Active"/>				
*Report Category ID:	<input type="text" value="ALLUSER"/> All PeopleSoft User				
Object Owner ID:	<input type="text" value="PeopleTools"/>				
*Template Type:	<input type="text" value="RTF"/>				
Retention Days:	<input type="text"/>				
Registered Date/Time:			Registered By:		
Updated Date/Time:			Updated By:		
Download:	Data Schema		Sample Data		

On the second page of the Report Definition, take note of the Template ID. You can change it if you desire.

For the Status, change it to be Active, then upload the RTF file. Then select SAVE.

Definition **Template** Output Properties Security Bursting

Report Name: XML_VCHR

Template Find | View All First

Template ID: XML_VCHR_1 **Default Template**

Description:

***Language Code:** English **Channel:**

Template Files Find | View All First

Effective Date: 06/06/2012

***Status:** Active

Template File: Sample_1.rtf **Use At**

Building a Page

For our test, we are going to modify the AP Voucher Entry page in Financials. You can also build your own page and use the same coding method we will demonstrate below.

We are going to modify the button found on the subpage VCHR_HDR_SBP. The field is VCHR_PANELS_WRK.AP_INV_PRINT_PB, but the code is on the Component VCHR_EXPRESS.

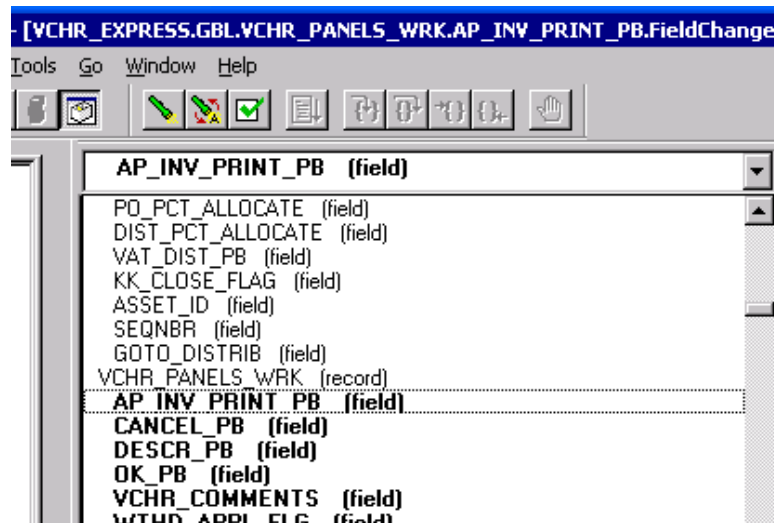


Business Unit: US001 Invoice No: 1001 Action:

Voucher ID: 00000081 Invoice Date: 09/12/2003

Voucher Style: Regular Voucher Accounting Date: 09/12/2003 Tax Exempt

Open the component VCHR_EXPRESS, and click the STRUCTURE tab to load the peoplecode. Right click and choose "View PeopleCode". Find the record VCHR_PANELS_WRK and the field AP_INV_PRINT_PB.



We will comment out the delivered code, which uses an App Package and Rowset processing.

```
/*import AP:Voucher:InvoicePrint;

Local AP:Voucher:InvoicePrint &oInvPrint = create AP:Voucher:InvoicePrint("VOUCHER");

&oInvPrint.PrintInvoice();
*/
```

Below is the new code you can use. This code is also in an attached TXT file on the CD.

```
/*Taken from http://www.pscustomizer.com/peoplesoft-examples/xmlp/xmlp-in-pc*/
/* !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!! */
/* You must reference an application package at the beginning of the event */
/* !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!! */
import PSXP_RPTDEFNMANAGER:*;

Local Record &rcdQryPrompts;
Local string &LanguageCd, &MyTemplate, &MyReportName, &OutFormat, &BusinessUnit, &VoucherID;
Local date &AsOfDate;

/* !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!! */
/* Set XML Publisher report required parameters                               */
/* !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!! */
&LanguageCd = "ENG";
&AsOfDate = %Date;
&OutFormat = "PDF";

/* !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!! */
/* Create a PDF using XML Report Definition MY_XMLP_DEMO                     */
/* !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!! */
&MyReportName = "XML_VCHR";
&MyTemplate = "XML_VCHR_1";

/* !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!! */
/* Get the values from the page */
/* !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!! */
&BusinessUnit = VOUCHER.BUSINESS_UNIT;
&VoucherID = VOUCHER.VOUCHER_ID;

/* !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!! */
/* Declare and Instantiate (construct) your Report Definition Object         */
/* !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!! */
Local PSXP_RPTDEFNMANAGER:ReportDefn &oReportDefn = create
PSXP_RPTDEFNMANAGER:ReportDefn(&MyReportName);

/* !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!! */
/* Get a handle on your Report Definition                                   */
/* !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!! */
&oReportDefn.Get();

/* !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!! */
/* Since there is a prompt to the query used in this report, you need to    */
/* provide the value for the prompt                                         */
/* !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!! */
&rcdQryPrompts = &oReportDefn.GetPSQueryPromptRecord();
If Not &rcdQryPrompts = Null Then
    &oReportDefn.SetPSQueryPromptRecord(&rcdQryPrompts);
    /* !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!! */
    /* Provide a value to the State Prompt                                   */
    /* !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!! */
    &rcdQryPrompts.BUSINESS_UNIT.Value = &BusinessUnit;
    &rcdQryPrompts.VOUCHER_ID.Value = &VoucherID;
End-If;

/* !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!! */
/* Kick of the report process                                               */
/* !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!! */
&oReportDefn.ProcessReport(&MyTemplate, &LanguageCd, &AsOfDate, &OutFormat);
```



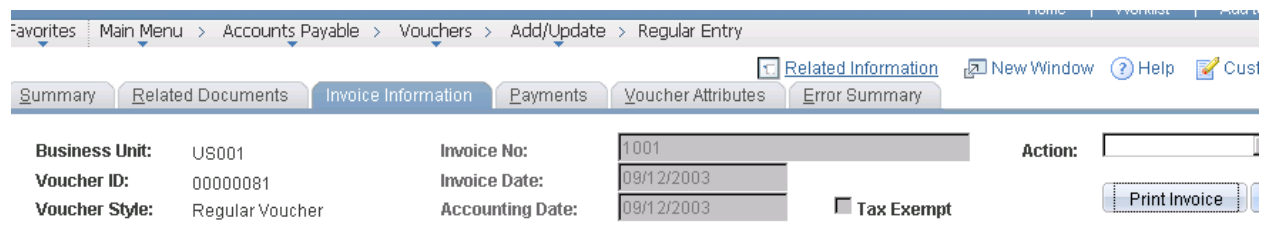
```

/* !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!! */
/* CommitWork must be called prior to displaying the output since the      */
/* application package performed work and SQL statements. If you do        */
/* not commit the work performed to this point you will receive an        */
/* error like "Think-time PeopleCode event (ViewAttachment), but a SQL     */
/* update has occurred in the commit interval. (2, 148)"                  */
/* !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!! */
CommitWork();

/* !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!! */
/* Display Report to the user                                              */
/* !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!! */
&oReportDefn.DisplayOutput();

```

Now you are ready to test the report. Navigate to the Voucher and open a voucher id. Then navigate to the Invoice Information tab and click "Print Invoice".

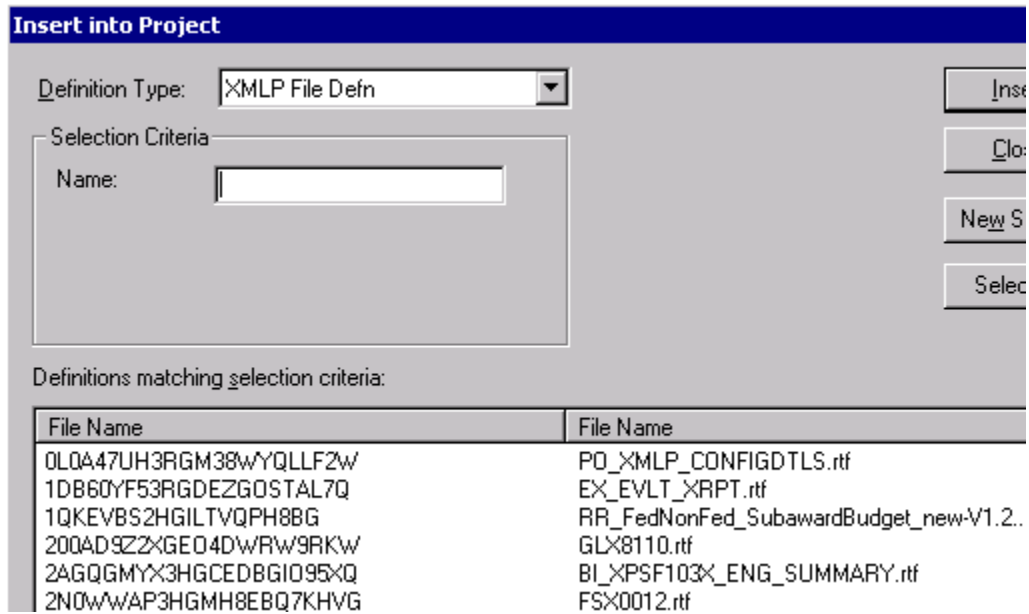


Migration Concerns

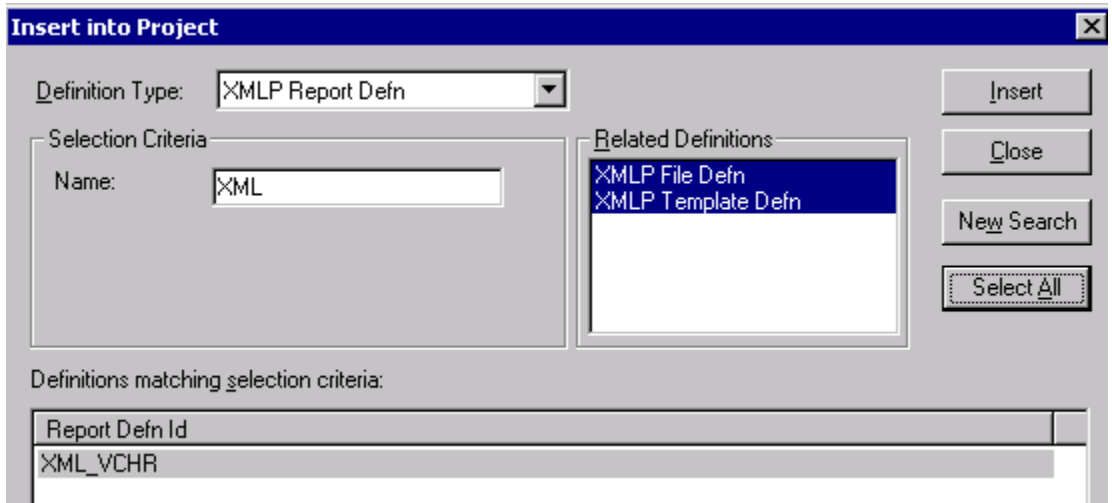
When migrating, you will need the following items in your project:

- XMLP Data Source Defn
- XMLP File Defn
- XMLP Report Defn
- XMLP Template Defn
- Query
- PeopleCode on Pushbutton

The tricky part is finding the XMLP File Defn. The File you attach to the Report Definition has an algorithmic type filename.



One trick is to use the “Related Items” when adding the Report Definition in App Designer.



You can also query the PSFILEDEFN table where USERFILENAME LIKE 'XYZ%'

Advanced Template Building

To take XML Publisher to the next level, this section will discuss advanced features in building your template. In addition, having expert Word training will also assist in producing a quality product.

Tables

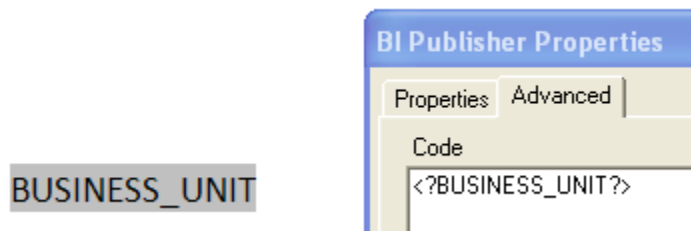
Use the menu Insert → Table Wizard. Once you are done, look for the special characters that mark the start and finish of the WHILE loop.

IF Conditioning

These come in handy, especially when you want the label for a field to be hidden when there is no data present.

Headings and Footers

You cannot drag and drop the fieldnames on Word's Header or Footer sections. But you can manually write the fieldname using the XMLP codes. For a field called BUSINESS_UNIT, use `<?BUSINESS_UNIT?>`. The fieldnames have given me trouble when I do not match the case; one easy solution is to drag the field to the body of the report, then click on the field and find the code under the Advanced tab.



Bar Codes

You need to load the font on both the local desktop and server. Next, you pick the field and change the font to be the Bar Code font. The final piece is you need to reference the bar code in both the local and server xdo.config file.

Disclaimer

Oracle and PeopleSoft are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners, such as Microsoft. David Vandiver does not represent Oracle, PeopleSoft, or Microsoft. The user reading this document takes full ownership of any consequences, be they ill or for good, and may not hold David Vandiver liable for the contents of this document, especially in the event of your database becoming corrupt or worse, truncated. If there are any further questions, the answer is 42.