

Oracle Apps XML Publisher Basic

Lesson 00

People matter, results count.



Copyright © Capgemini 2015. All Rights Reserved 1

©2016 Capgemini. All rights reserved.
The information contained in this document is proprietary and confidential. For
Capgemini only.

Document History

Date	Course Version No.	Software Version No.	Developer / SME	Change Record Remarks
11-Mar-2015	1.0		BU/Amit Sali	Content Creation
4-April-2015			CLS Team	Review



Copyright © Capgemini 2015. All Rights Reserved 2

Course Goals and Non Goals

- Course Goals

- Creating a xml publisher report by using rdf report
- Registration xml publisher report in oracle Application

- Course Non Goals

- In detail features of XML Publisher



Copyright © Capgemini 2015. All Rights Reserved 3

Pre-requisites

- Knowledge of SQL/PLSQL
- Knowledge of Unix
- Basic knowledge of reports
- Oracle Application fundamentals



Copyright © Capgemini 2015. All Rights Reserved 4

Intended Audience

- Programmers and Analysts



Day Wise Schedule

- Day 1

- Lesson 1: Introduction to XML Reports

- Day2

- Lesson 2: Template Creation and Registration in oracle Application



Copyright © Capgemini 2015. All Rights Reserved 6

Table of Contents

- Lesson 1: Introduction to XML Reports
 - 1.1 Introduction
 - 1.2 Process Overview
 - 1.3 XML Publisher concepts
 - 1.4 Report Features
- Lesson 2: Template Creation and Registration in oracle Application
 - 2.1 Process to Generate XML Tag
 - 2.2 Rtf Design and registration
 - 2.3 Data definition and Template Design
 - 2.4 Run Report with RTF Template



Copyright © Capgemini 2015. All Rights Reserved 7

References

- Student Guide
 - All slides presented during lecture along with explanation
- Lab Guide
 - Hands-on lab exercises with sample solutions
- CDROM
 - Pre-requisite Files / Scripts for Assignments
 - Reference Material



Copyright © Capgemini 2015. All Rights Reserved 8

Next Step Courses (if applicable)

- Detail understanding of XML Publisher
- XML Busting



Other Parallel Technology Areas

- NA



Copyright © Capgemini 2015. All Rights Reserved 10

Xml Publisher/BI Publisher

(Oracle XML Publisher
release 5.6.3)

Document History

Date	Course Version No.	Software Version No.	Developer / SME	Change Record Remarks
21-Jan-2015	1.0	5.6.3		Initial Version



Copyright © Capgemini 2015. All Rights Reserved 2

Course Goals

- Course Goals

- Goal from this document/presentation is to understand, utilize, usability and accessibility on XML Publisher. XML Publisher is a standalone Oracle product and an E-Business Suite integrated application module.
- It explains how to navigate, query, extract data and run reports using the user interface (UI) of Oracle Applications.
- It provides a new approach to report design and publishing by Integrating desktop word processing tools with existing E-Business Suite data reporting.
- Document covers the fundamental features of XML Publisher, also known as Oracle Business Intelligence Publisher (BI Publisher), in an Oracle E-Business Suite environment. You learn how to create templates and reports as well as how to publish your reports.



Copyright © Capgemini 2015. All Rights Reserved 3

Intended Audience

- Technical Programmer



Table of Contents

- Lesson 1: Chapter 1
 - 1.1 Introduction
 - 1.2 Process Overview
 - 1.3 XML Publisher concepts
 - 1.4 Report Features



Copyright © Capgemini 2015. All Rights Reserved 5

Introduction

- Oracle XML Publisher is a template-based publishing solution delivered with the Oracle E-Business Suite.
- The flexibility of XML Publisher is a result of the separation of the presentation of the report from its data structure.
- At run time, XML Publisher merges the custom templates with the concurrent request data extracts to generate output in PDF, HTML, RTF and EXCEL.
- XML Publisher provides the Template Manager to register and maintain report templates and their data sources.
- With a single template, it can generate reports in many formats (PDF, RTF, Excel, HTML, and so on) in many languages



Copyright © Capgemini 2015. All Rights Reserved 6

Introduction

- Template Builder is tightly integrated with MS Word that allows you to perform the following actions:
- Insert data fields into RTF Template
- Insert tables, forms, charts and cross tabs
- Preview your template in multiple outputs
- Perform calculations multiple fields within the Template



Copyright © Capgemini 2015. All Rights Reserved 7

XMLP Technology

- Open standard technologies
 - Pure Java
 - Pluggable data in XML
 - Output formats in PDF, PS, RTF, HTML...
 - Support for Internet Printing Protocol, WebDAV, Internet Fax Protocol, SMTP, etc...



Process Overview

- XML Publisher, which is also called Oracle Business Intelligence Publisher (BI Publisher), is a template-based reporting tool that leverages standard technologies for data extraction and display.
- Business users can build the layout and rules for the report themselves by using common desktop tools.
- Development engineers can concentrate on extracting data in the most efficient manner. Deployment time is faster.
- Also, business users with minimal training can design templates and reports based on the data sources provided by engineers, thus greatly reducing the cost of ownership.
- Creating customized reports using XML Publisher can be divided into two phases: Design Time and Runtime.



Copyright © Capgemini 2015. All Rights Reserved 9

Process Overview

- Design Time
- Register the E-Business Suite report as a Data Definition in the Template Manager. Create a Data Definition in the Template Manager for E-Business Suite reports that you wish to customize using XML Publisher. When you create the Data Definition, the Data Definition Code must match the E-Business Suite report short name.



Copyright © Capgemini 2015. All Rights Reserved 10

Process Overview

- Design your template. Your template files can be either in Rich Text Format (RTF) or Portable Document Format (PDF). RTF is a specification used by many word processing applications, such as Microsoft Word. You design the template using your desktop word processing application and save the file as an RTF file type (.rtf extension). Insert basic markup tags to the document to prepare it for merging with the XML data. XML Publisher recognizes the formatting features that you apply and converts them to XSL .Use Adobe Acrobat to apply markup tags to your custom-designed or downloaded PDF template .



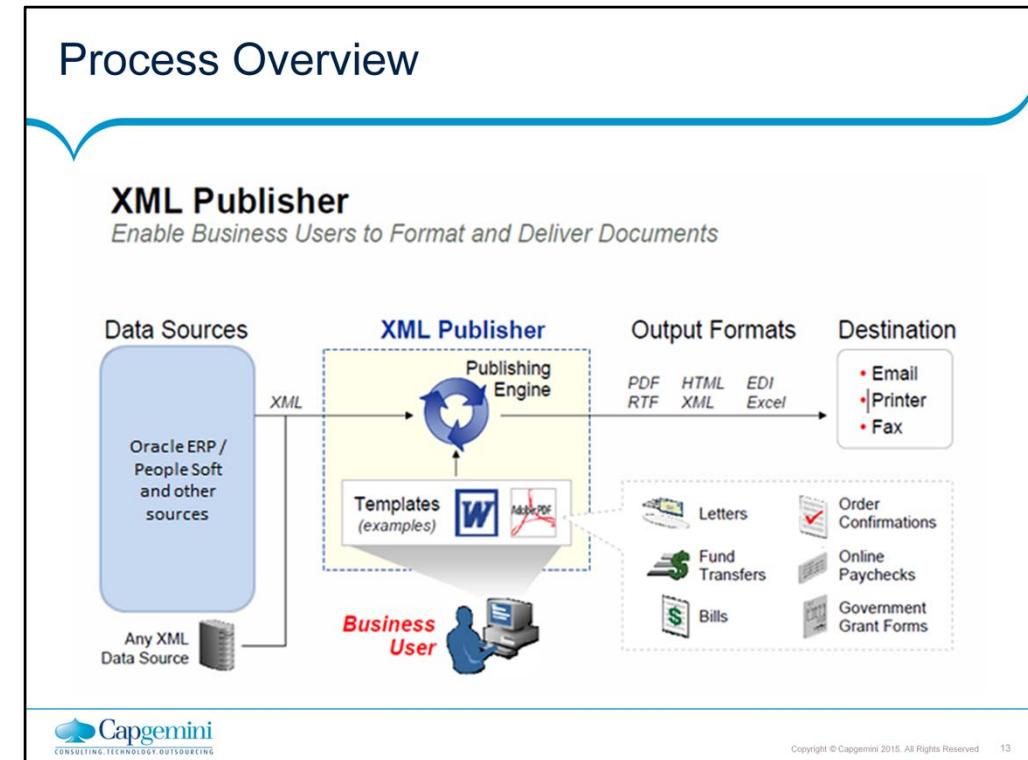
Copyright © Capgemini 2015. All Rights Reserved 11

Process Overview

- Create a Template in the Template Manager for your template design file. When you create the template in the Template Manager, you register and upload your template design files. The Template must be assigned to the Data Definition Code of the E-Business Suite report with which it will be merged.
- Runtime
- Set the concurrent program to generate XML.
- Run the concurrent program using standard request submission to obtain the XML output.
- Run the XML Publisher Concurrent Request.



Copyright © Capgemini 2015. All Rights Reserved 12



Add the notes here.

XML Publisher Concept

- XML Publisher breaks the three components apart and treats them separately at design time and at runtime the three are brought back together by XML Publisher to generate the final formatted, translated output.
- This model is far more flexible i.e. a single data definition can support multiple layouts, multiple translations can be applied at runtime to generate translated output. This leads to a reduction in maintenance costs for all concerned .

```
graph LR; DL[Data Logic] --> XPU[XML Publisher]; L[Layout] --> XPU; T[Translation] --> XPU; XPU --> PO[Report output]; XPU --> FO[File Output];
```

The diagram shows three colored boxes on the left: orange for Data Logic, dark blue for Layout, and yellow for Translation. Arrows point from each of these boxes to a central gear icon labeled "XML Publisher". From the gear, two arrows emerge: one pointing to a stack of three document icons labeled "Report output" and another pointing to a row of five small icons representing different file formats: PDF, Excel, Word, XML, and EFT.

Capgemini
CONSULTING TECHNOLOGY OUTSOURCING

Copyright © Capgemini 2015. All Rights Reserved 14

Add the notes here.

XML Publisher Concept

- Data Logic: Data extracted from the database and converted into a XML string.
- Layout: The layout templates to be used for final output are stored and managed in the Template manager.
- Translation: The translation handler will manage the translation that is required at run time.



Copyright © Capgemini 2015. All Rights Reserved 15

Report Features

- How do I get my XML data?
- eBusiness Suite
- Oracle Reports
 - A concurrent program designed to generate XML
- Data Template
 - PeopleSoft (People Tools 8.48)
 - PS Query
- People Code XML Doc
- Web Services
 - JDE (Enterprise One 8.96)
 - Enterprise One Reports

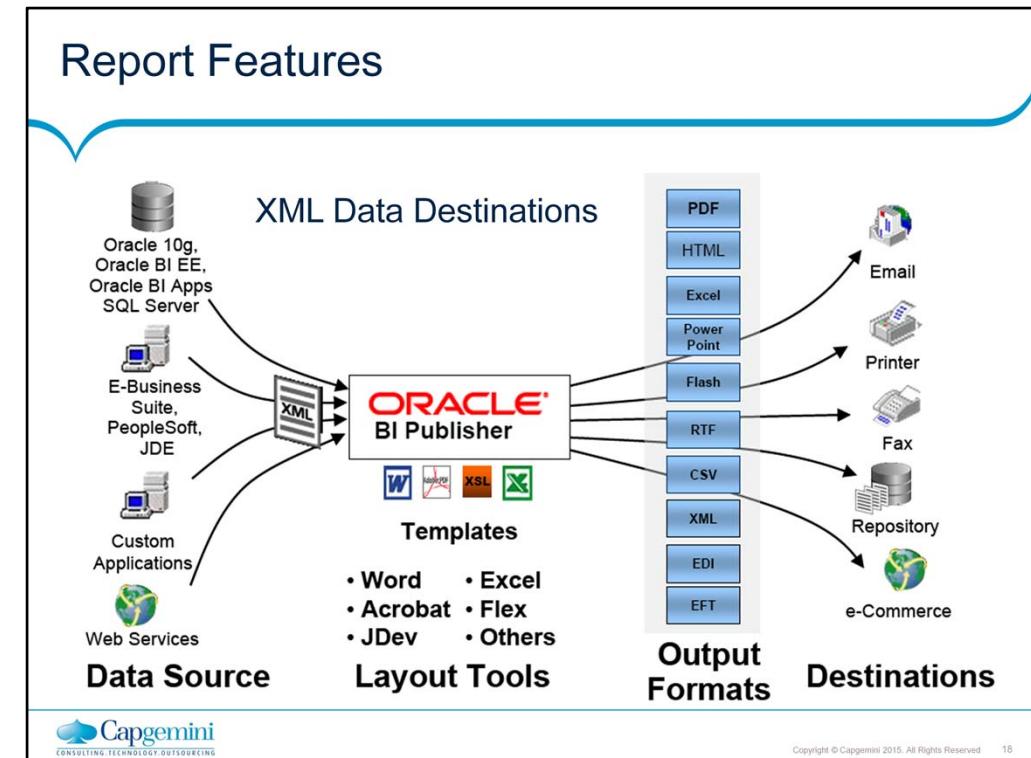


Copyright © Capgemini 2015. All Rights Reserved 16

Report Features

- Integrated Document Management
- Oracle XML Publisher
- Authoring
- Managing
- Delivering
- Your Business Documents
- Meet Business Requirements
- Remove Complexity
- Reduce Maintenance Cost
- Reduce Total Cost





Data Source

XML Publisher can accept data from a host of data providers, it uses XML data so if a source can generate XML then XML Publisher can be integrated into the system.

Templates

Layout templates are built using familiar desktop applications such as MSWord and Adobe Acrobat. Plugins are provided to help the report designer to build the layout they want in their favored application. Customers are not tied to these applications, XML Publisher has a W3C XSLFO implementation at its heart and there are many visual XSLFO editors available that can generate templates for use with XML Publisher.

Outputs

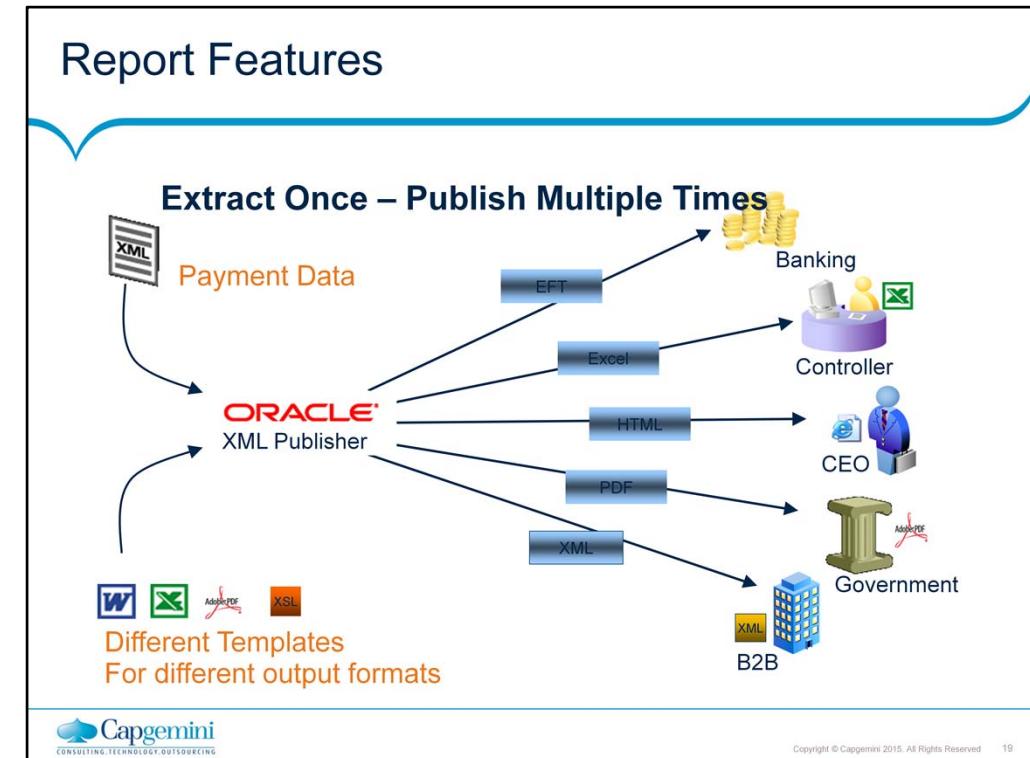
XML Publisher is able to generate multiple output formats from a single data source and template:

- PDF
- RTF
- Excel
- HTML
- XML
- EDI
- EFT

Destinations

The generated output can then be delivered to many delivery channels including:

- Printer
- Email
- FAX
- WebDAV (FilesOnline)
- HTTP/S
- FTP



Extract Once – Publish Multiple Times

XML Publisher has the ability to publish multiple times using the same data source. For instance, a company can generate its payment data in a single data extraction program. From that they can generate:

Electronic Funds Transfer (EFT) to communicate to banks to make payments to suppliers

Excel – Accountants and controllers are able to get the payment data into Excel for further analysis

HTML – Payment data can be published to a company portal via HTML

PDF – Government reporting requires the PDF format for communication

XML – for B2B communication XML is fast becoming the de-facto standard

With the use of different layout templates XML Publisher simplifies the reporting architecture for customers.

Report Features

- Full range of charting capabilities
- Conditional formatting
- Report interactivity: filter, sort, hide columns
- Formulas and functions
- Multiple table/forms layout
- Watermark support
- Rich text formatting



Copyright © Capgemini 2015. All Rights Reserved 20

Report Features

- Design Tools
 - Use Microsoft Word or Adobe Acrobat to design layout
 - Use XMLP's Template Build plug-in to Microsoft Word
 - Font Support
 - Advanced font sub setting and embedding
 - Support for external fonts, such as barcodes and MICR
- Document Generation
 - Output formats: PDF, RTF, HTML, Excel, XML, PPT
- Language Support
 - No dependency on installed languages or character set
 - Generate XLIFF file for each translation
- Security
 - Support PDF security levels – Read Only/Edit, Copy Text, Printable, Password Protection



Copyright © Capgemini 2015. All Rights Reserved 21

Report Features

- Delivery
- Supports delivery via email
- Redeliver documents
- Supports bursting of one report to multiple destinations
- Localization and Translation
 - Translation allows exchange of one language to the next
 - Localization goes beyond in excluding/including additional data elements



Copyright © Capgemini 2015. All Rights Reserved 22

References

- http://docs.oracle.com/cd/B34956_01/current/acrobat/120xdoig.pdf
- https://docs.oracle.com/cd/B40089_10/current/acrobat/120xdorg.pdf
- <http://www.oracle.com/technetwork/middleware/bi-publisher/overview/bipublisherbestpractices2012-1869524.pdf>



Questions ?

- Xml Publisher is which type of reporting tool ?
- What are the output format of Xml Publisher ?
- What are the advantages of Xml Publisher ?
- Describe different features of Xml/BI publisher ?



Copyright © Capgemini 2015. All Rights Reserved 24

Xml Publisher/BI Publisher

(Oracle XML Publisher
release 5.6.3)

Document History

Date	Course Version No.	Software Version No.	Developer / SME	Change Record Remarks
21-Jan-2015	1.0	5.6.3		Initial Version



Copyright © Capgemini 2015. All Rights Reserved 2

Intended Audience

- Technical Programmer



Table of Contents

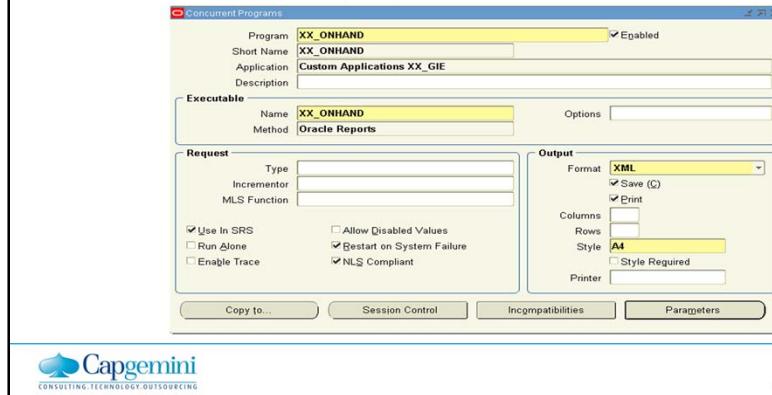
- Lesson 2: Chapter 2
 - 2.1 Process to Generate XML Tag
 - 2.2 Rtf Design and registration
 - 2.3 Data definition and Template Design
 - 2.4 Run Report with RTF Template



Copyright © Capgemini 2015. All Rights Reserved 4

Process to Generate XML Tag

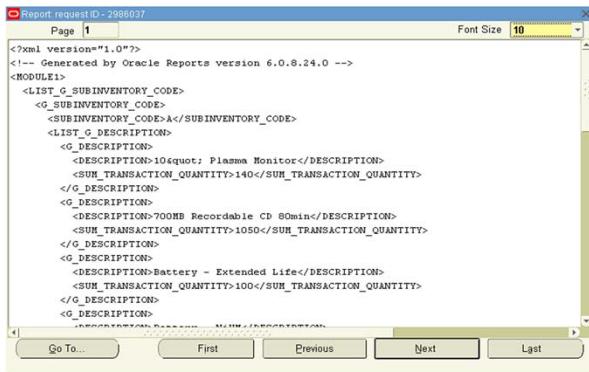
- We can generate Xml Tags by Registering an Oracle reports or PI/Sql stored Procedure, then register it as Concurrent program with output format type as “XML”
- Execute the concurrent program to generate XML tags as Output file
- See in example for (.rdf)



Copyright © Capgemini 2015. All Rights Reserved 5

Process to Generate XML Tag

- Run the Concurrent Program
- As the output format is set as XML the xml code will be displayed
- Save the output as an XML file ONHAND.xml



The screenshot shows a report window titled "Report request ID - 2906037" with "Page 1" selected. The font size is set to 10. The XML code displayed is:

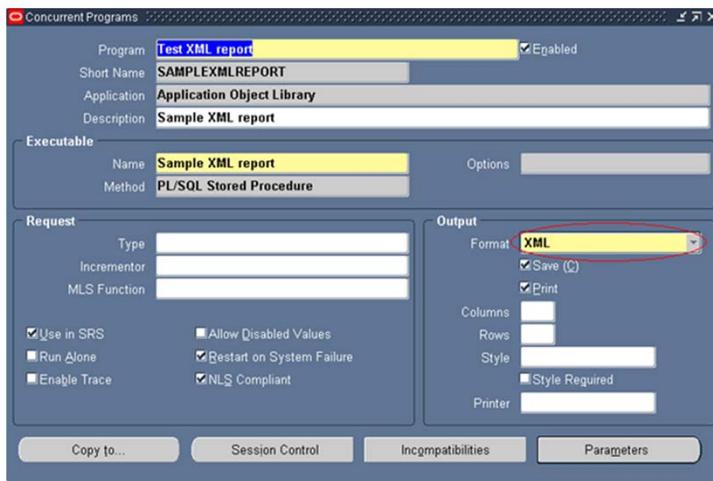
```
<?xml version="1.0"?>
<!-- Generated by Oracle Reports version 6.0.8.24.0 -->
<MODULE>
  <LIST_G_SUBINVENTORY_CODE>
    <G_SUBINVENTORY_CODE>
      <SUBINVENTORY_CODE>a</SUBINVENTORY_CODE>
      <LIST_G_DESCRIPTION>
        <G_DESCRIPTION>
          <DESCRIPTION>10<quot; Plasma Monitor</DESCRIPTION>
          <SUM_TRANSACTION_QUANTITY>140</SUM_TRANSACTION_QUANTITY>
        </G_DESCRIPTION>
        <G_DESCRIPTION>
          <DESCRIPTION>700M Recordable CD 80min</DESCRIPTION>
          <SUM_TRANSACTION_QUANTITY>1050</SUM_TRANSACTION_QUANTITY>
        </G_DESCRIPTION>
        <G_DESCRIPTION>
          <DESCRIPTION>Battery - Extended Life</DESCRIPTION>
          <SUM_TRANSACTION_QUANTITY>100</SUM_TRANSACTION_QUANTITY>
        </G_DESCRIPTION>
        <G_DESCRIPTION>
```



Copyright © Capgemini 2015. All Rights Reserved 6

Process to Generate XML Tag

- See in example for (Pl/Sql Stored Procedure)



Copyright © Capgemini 2015. All Rights Reserved 7

Process to Generate XML Tag

- Run the Concurrent program from responsibility and will see the XML output appearing in the View output and save the output as sample_report.xml on your computer



The screenshot shows a web browser displaying an XML document. The URL in the address bar is http://igatehydap05.igate.com:8005/OA_CGI/FNDWR.... The XML content is as follows:

```
<?xml version="1.0" ?>
- <EMPINFO>
  - <P_EMP>
    <EMPID>718472</EMPID>
    <EMPNAME>Ajay</EMPNAME>
    <SALARY>20000</SALARY>
    <DEPT>Oracle</DEPT>
    <DESG>Software Engineer</DESG>
    <ADDRESS>Chennai</ADDRESS>
    <CONTACT>7411386021</CONTACT>
  - </P_EMP>
  - <P_EMP>
    <EMPID>718473</EMPID>
    <EMPNAME>Gopi</EMPNAME>
    <SALARY>20000</SALARY>
    <DEPT>Oracle</DEPT>
    <DESG>Software Engineer</DESG>
    <ADDRESS>Hyderabad</ADDRESS>
    <CONTACT>7411386022</CONTACT>
  - </P_EMP>
  - <P_EMP>
    <EMPID>718471</EMPID>
    <EMPNAME>Vijay</EMPNAME>
    <SALARY>20000</SALARY>
    <DEPT>Oracle</DEPT>
    <DESG>Software Engineer</DESG>
    <ADDRESS>Bangalore</ADDRESS>
    <CONTACT>7411386023</CONTACT>
  - </P_EMP>
</EMPINFO>
```



Copyright © Capgemini 2015. All Rights Reserved 8

Rtf Design and Registration

- BI Publisher supports several types of templates. RTF templates are the most common and provide a wide range of output formats. You can use the BI Publisher Template Builder, which is an Add-in to Microsoft Word, to build an RTF template.
- For rtf design we need to import sample_report.xml to Word file.
- Open the MS Word and you will see menu item called Add-Ins



- Choose Data > Load XML Data and select the file sample_report.xml.
- NOTE: Upgrade Version It may differ in BIP/XML Publisher tool menu.

PNG image



Copyright © Capgemini 2015. All Rights Reserved 9

Rtf Design and Registration

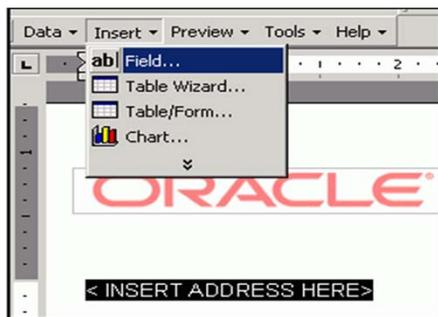
- When the "Data loaded successfully" message appears, click OK..



- There are two types of methods to put the xml tags In the template
- Basic RTF Method: Use any word processing application that supports RTF to design a template using XML Publisher's simplified syntax
- Form Field Method: It allows you to place the syntax in hidden form fields, rather than directly into the design of template

Rtf Design and Registration

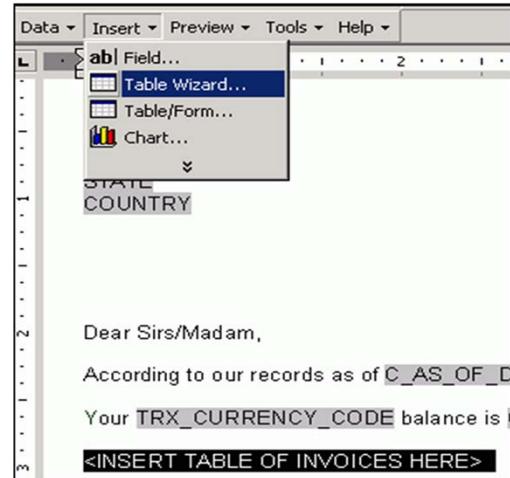
- Insert Form Fields
- Select <INSERT ADDRESS HERE> - xml tag and
- select Insert > Field from the BI Publisher menu.



- So here we are just importing a field .

Rtf Design and Registration

- Create a Table wizard
- Select the xml tag <INSERT TABLE OF INVOICES HERE> -in the layout template
- Select Insert > Table Wizard from the BI Publisher menu.



Copyright © Capgemini 2015. All Rights Reserved 12

Rtf Design and Registration

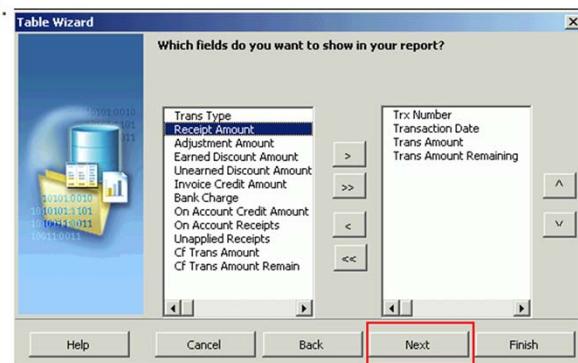
- In the Table Wizard window, select the Table option and click Next



Copyright © Capgemini 2015. All Rights Reserved 13

Rtf Design and Registration

- Select the fields shown in the image below and move them to the right column. Click Next.



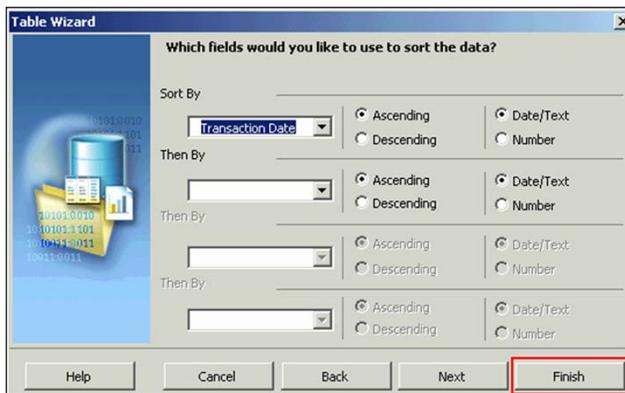
- For Grouping and sorting Field, select the item from the drop-down list. Click Next



Copyright © Capgemini 2015. All Rights Reserved 14

Rtf Design and Registration

- In the Sort By window, select Transaction Date from the drop-down list. Let the sort be Ascending order, which is the default. Click Finish.



Rtf Design and Registration

- Your template should appear similar to the image below. Use the Microsoft Word features to modify the look and feel, and adjust the columns of the template.
- Save the below Template with extension (.rtf). Lets say as “Balance Letter-Template.rtf”

CUSTOMER_NAME
ADDRESS_LINE1
ADDRESS_LINE3
ADDRESS_LINE4
CITY
STATE
COUNTRY

Dear Sirs/Madam,

According to our records as of **C_AS_OF_DATE_DISPLAY** we show the following open balances.

Your **TRX_CURRENCY_CODE** balance is **C_INV_OPEN_BALANCE**, made up as follows:

Trx Number	Transaction Date	Trans Amount	Trans Amount Remaining
TRX_NUMBER	TRANSACTION_DATE	TRANS_AMOUNT	TRANS_AMOUNT_REMAINING_E

Please confirm the above information.

Thanking you in advance, we remain.

Yours faithfully,



Copyright © Capgemini 2015. All Rights Reserved 16

Data Definition and Template Design

- Now we have to register the rtf in Xml Publisher Responsibility, for that we need to create Data definition associate with Data template.
- Data definition: It is a part of EBS template Manager that allows users to associate a Data Template, Bursting Control File and XML Preview Data for a Concurrent Program. Simply, it registers the data source that BI Publisher will use to merge with a template and create a finished report. Data definition having below fields
 - Name: Enter a user-friendly name for your data definition.
 - Code: The data definition Code must match the concurrent program short name of the report
 - Application: Select the report's application from the LOV.



Copyright © Capgemini 2015. All Rights Reserved 17

Data Definition and Template Design

- Start Date: Enter the date from which the data definition will be active (System Defaults the Sysdate).
- End Date You cannot delete data definitions from the Template Manager. To make the data definition inactive, enter an end date.
- Data Template If you are using an XML Publisher data template to generate the data for this data definition, use the Add File button to upload your data template.



Copyright © Capgemini 2015. All Rights Reserved 18

Data Definition and Template

- Data Template is the method by which you communicate your request for data to the data engine. It is an XML document whose elements collectively define how the data engine will process the template to generate the XML.



Copyright © Capgemini 2015. All Rights Reserved 19

Data Definition and Template

■ Data Definition:

- To publish your report, you must register the data source that would be merged with the template. When you register your layout template file (in the next section), you must assign to it a data definition that exists in the Template Manager. This associates the two at run time. Multiple templates can use the same data definition.
- To create a data definition, perform the following steps:
 - Log in to E-Business Suite and click XML Publisher
 - Click the Data Definitions link and you will be on the Data Definition tab for XML Publisher. Click the Create Data Definition button to create a new data definition

The screenshot shows the Oracle XML Publisher interface for managing data definitions. At the top, there's a navigation bar with links for Home, Logout, Preferences, Help, and Diagnostics. Below that, a sub-menu for Data Definitions is selected. The main area has a search form with fields for Name, Application, Code, and Active status, along with a Go button and a search icon. A prominent red box surrounds the 'Create Data Definition' button. Below the search form is a table with columns: Name, Application, Start Date, End Date, and Update. The Capgemini logo is at the bottom left, and the page footer includes copyright information: Copyright © Capgemini 2015. All Rights Reserved 20.

Data Definition and Template Design

- Under Create Data Definition, fill the form as shown below. Use the correct “short name of the Concurrent Program as Data Definition Code” –let as example “ARXCOBLX” to correctly link this data definition to the underlying Concurrent Program. For the Name field, you can use whatever you like. Lets use “Customer Balance Data Definition.” Click Apply.

The screenshot shows the 'Create Data Definition' dialog box in Oracle XML Publisher. The 'General' tab is selected. The 'Name' field contains 'Customer Balance Data'. The 'Code' field contains 'ARXCOBLX'. The 'Application' field is set to 'Receivables'. The 'Start Date' field is set to '08-Mar-2008'. The 'End Date' field is empty. The 'Cancel' and 'Apply' buttons are at the bottom right, with 'Apply' being highlighted with a red box.

Data Definition and Template Design

- See a confirmation message. Now you have successfully created the data definition for the Customer Balance Letter.

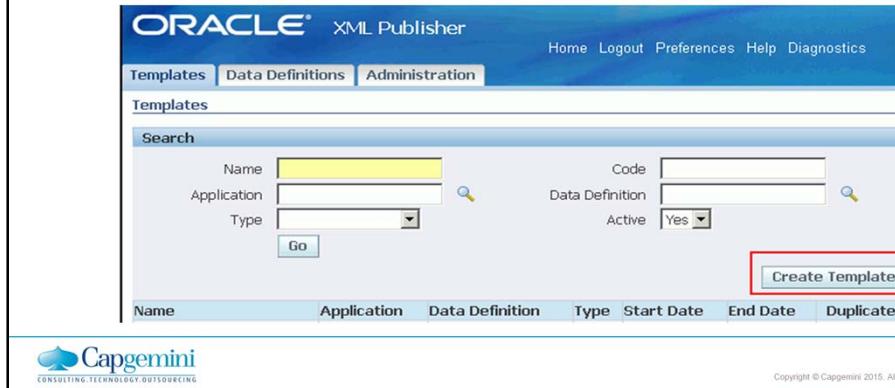
The screenshot shows a user interface for 'Data Definition and Template Design'. At the top, there are three tabs: 'Templates', 'Data Definitions', and 'Administration'. The 'Data Definitions' tab is currently selected, indicated by a blue background and white text. Below the tabs, a breadcrumb navigation shows 'Data Definitions >'. A prominent message box displays a green checkmark icon followed by the text 'Confirmation' and 'Data Definition Customer Balance Data Definition has been successfully created.' At the bottom of the message box is a blue link labeled 'View Data Definition: Customer Balance Data Definition'. The overall layout is clean and professional, typical of enterprise software interfaces.



Copyright © Capgemini 2015. All Rights Reserved 22

Data Definition and Template Design

- Once Data definition created successfully we have to map this to Template
- Following steps to create Template
- On the Template tabbed page for XML Publisher, click Create Template.



Data Definition and Template Design

- Enter the template name as Customer Balance Letter - Template. Select Application name as Receivables using the Lookup icon. Enter code as ARXCOBLX.

* Name	Customer Balance Letter	* Code	ARXCOBLX
* Application	Receivables	* Data Definition	
* Type		* Start Date	11-Mar-2008

Data Definition and Template Design

- Click the Lookup icon next to the Data Definition field, and then click Go. Select the Customer Balance Data definition option from the list and click Select.
- Select the template type as RTF from the Type drop-down list.

The screenshot shows the 'Create Template' dialog box with the 'General' tab selected. The 'Name' field contains 'Customer Balance Letter'. The 'Application' field is set to 'Receivables'. The 'Type' dropdown menu is open, with 'RTF' selected and highlighted by a red box. Other fields include 'Code' (ARXCBLX), 'Data Definition' (Customer Balance Data d), 'Start Date' (11-Mar-2008), 'End Date' (empty), and 'Subtemplate' (empty). Buttons for 'Cancel' and 'Apply' are visible at the bottom right.



Copyright © Capgemini 2015. All Rights Reserved 25

Data Definition and Template Design

- We need to upload the template that we created in Microsoft Word, browse and open the file called Balance Letter - Template.rtf



Data Definition and Template Design

- Select the Language as English.

Search and Select: Language

Cancel Select

Search
To find your item, select a filter item in the pulldown list and enter a value in the text field, then select the "Go" button.

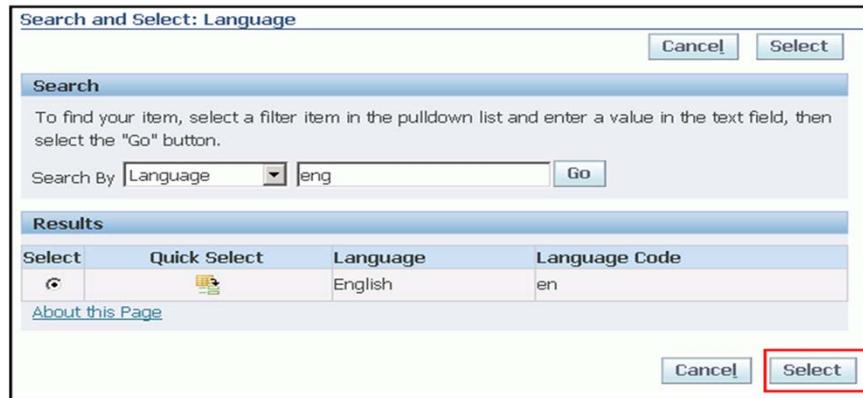
Search By Language eng Go

Results

Select	Quick Select	Language	Language Code
<input checked="" type="radio"/>		English	en

About this Page

Cancel Select



Copyright © Capgemini 2015. All Rights Reserved 27

Data Definition and Template Design

- Here is what your template form should look like. Click Apply. You should get a message that your template has been created successfully.

The screenshot shows the 'Template' dialog box with the following fields:

* Name: Customer Balance Letter	* Application: Receivables	* Type: RTF	* Code: ARXCOBLX	* Data Definition: Customer Balance Data
			* Start Date: 11-Mar-2008	End Date:
Description: [Empty Text Area]				
Template File				
* File: Balance Letter - Template.rtf				
* Language: English				
Territory: [Empty Text Area] <input type="checkbox"/> Translatable				
<input type="button" value="Cancel"/> <input style="border: 2px solid red; padding: 2px; margin-right: 10px;" type="button" value="Apply"/>				



Copyright © Capgemini 2015. All Rights Reserved 28

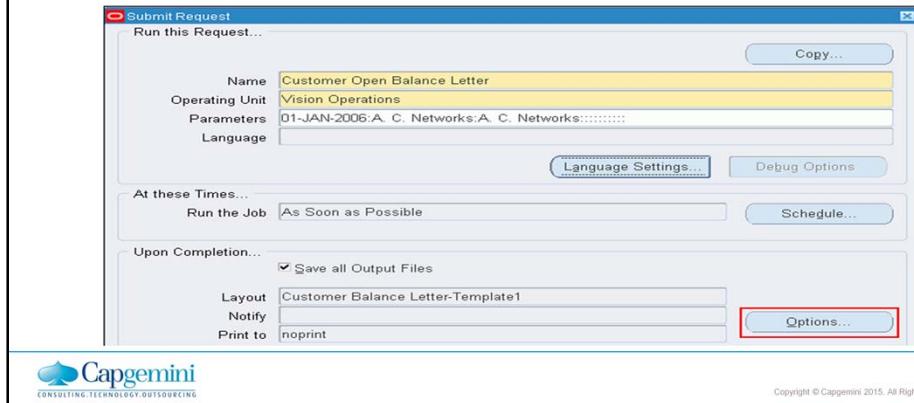
Run Report with RTF Template

- Navigate to Receivables, Vision Operations (USA) responsibility.
Click Run link from Control: Requests



Run Report with RTF Template

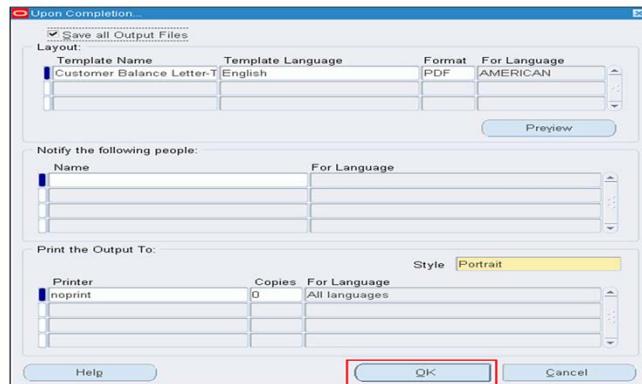
- Select “Customer Open Balance Letter” from the list and click OK
- Note that in the Layout field, the layout template file is listed. Click the Options button to see the other options that you can set after the report is completed.



Copyright © Capgemini 2015. All Rights Reserved 30

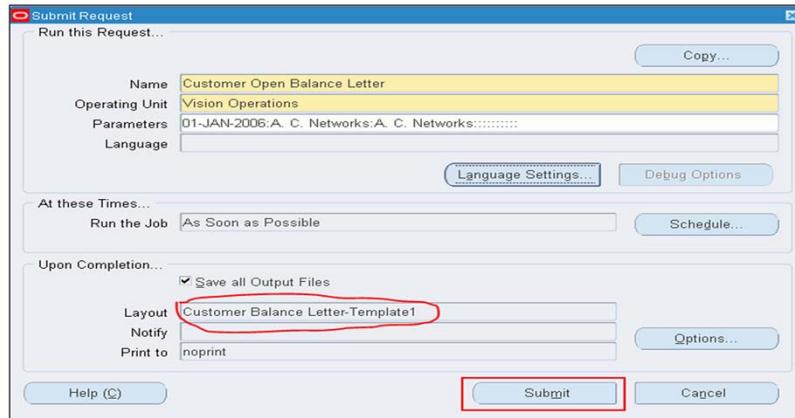
Run Report with RTF Template

- The Upon Completion.. Window appears. In this window, you can specify what actions can be taken after the job is executed. You can notify one or more people about the report. You can send the report to be printed.



Run Report with RTF Template

- Return to the Submit Request window. Click Submit to execute the request.



Copyright © Capgemini 2015. All Rights Reserved 32

Run Report with RTF Template

- Once request is success then we can see Report in out file.

The screenshot shows the Oracle BI Publisher Requests interface. At the top, there are three buttons: 'Refresh Data', 'Find Requests', and 'Submit a New Request...'. Below these buttons is a table with columns: Request ID, Name, Parent, Phase, Status, and Parameters. A single row is visible in the table, highlighted with a red border. The row contains the following data:

Request ID	Name	Parent	Phase	Status	Parameters
4369055	Customer Open Balance L	4369054	Completed	Normal	2006/01/01 00:00:00, A. C. Ne

At the bottom left is the Capgemini logo, and at the bottom right are the copyright information 'Copyright © Capgemini 2015. All Rights Reserved' and the page number '33'.

Run Report with RTF Template

- Click the View Output button to view the report in a browser. Note that the report is displayed with the layout in the template.

A. C. Networks 3405 East Bay Blvd. Provo UT US																																								
Dear Sirs/Madam,																																								
According to our records as of 01 January 2006 we show the following open balances.																																								
Your USD balance is \$515,752.16, made up as follows:																																								
<table border="1"><thead><tr><th>Transaction #</th><th>Transaction Date</th><th>Amount</th><th>Amount Due</th></tr></thead><tbody><tr><td>S10250</td><td>05-DEC-05</td><td>\$77,398.75</td><td>\$77,398.75</td></tr><tr><td>S10220</td><td>08-DEC-05</td><td>\$43,164.69</td><td>\$43,164.69</td></tr><tr><td>S10241</td><td>12-DEC-05</td><td>\$142,779.59</td><td>\$142,779.59</td></tr><tr><td>S10262</td><td>13-DEC-05</td><td>\$19,349.59</td><td>\$19,349.59</td></tr><tr><td>S10359</td><td>20-DEC-05</td><td>\$20,700.00</td><td>\$20,700.00</td></tr><tr><td>S10382</td><td>21-DEC-05</td><td>\$87,871.94</td><td>\$87,871.94</td></tr><tr><td>S10450</td><td>27-DEC-05</td><td>\$63,867.50</td><td>\$63,867.50</td></tr><tr><td>S10480</td><td>28-DEC-05</td><td>\$60,620.00</td><td>\$60,620.00</td></tr><tr><td colspan="2" style="text-align: right;">Total</td><td></td><td>\$515,752.16</td></tr></tbody></table>	Transaction #	Transaction Date	Amount	Amount Due	S10250	05-DEC-05	\$77,398.75	\$77,398.75	S10220	08-DEC-05	\$43,164.69	\$43,164.69	S10241	12-DEC-05	\$142,779.59	\$142,779.59	S10262	13-DEC-05	\$19,349.59	\$19,349.59	S10359	20-DEC-05	\$20,700.00	\$20,700.00	S10382	21-DEC-05	\$87,871.94	\$87,871.94	S10450	27-DEC-05	\$63,867.50	\$63,867.50	S10480	28-DEC-05	\$60,620.00	\$60,620.00	Total			\$515,752.16
Transaction #	Transaction Date	Amount	Amount Due																																					
S10250	05-DEC-05	\$77,398.75	\$77,398.75																																					
S10220	08-DEC-05	\$43,164.69	\$43,164.69																																					
S10241	12-DEC-05	\$142,779.59	\$142,779.59																																					
S10262	13-DEC-05	\$19,349.59	\$19,349.59																																					
S10359	20-DEC-05	\$20,700.00	\$20,700.00																																					
S10382	21-DEC-05	\$87,871.94	\$87,871.94																																					
S10450	27-DEC-05	\$63,867.50	\$63,867.50																																					
S10480	28-DEC-05	\$60,620.00	\$60,620.00																																					
Total			\$515,752.16																																					
Please confirm the above information.																																								
Thanking you in advance, we remain.																																								



Copyright © Capgemini 2015. All Rights Reserved 34

References

- http://docs.oracle.com/cd/B34956_01/current/acrobat/120xd0ig.pdf
- https://docs.oracle.com/cd/B40089_10/current/acrobat/120xdorg.pdf
- <http://www.oracle.com/technetwork/middleware/bi-publisher/overview/bipublisherbestpractices2012-1869524.pdf>



Copyright © Capgemini 2015. All Rights Reserved 35

Questions ?

- What are different methods to generate Xml File ?
- Describe steps to generate Xml File ?
- What is Data Definition ?
- Describe steps to design Data Definition and Template ?



XML Publisher Lab Book

Document Revision History

Date	Revision No.	Author	Summary of Changes
22-01-2015	1.0	BU/Amit Sali	Content Creation

Table of Contents

Document Revision History	2
Table of Contents.....	3
Getting Started	4
Overview.....	4
Setup Checklist for EAS Oracle Application.....	4
Instructions	4
XML Publisher Basic	5
Introduction.....	5
Process Overview.....	6
XML Publisher Concept.....	7
Lab1: Creating a xml publisher report by using rdf report	8
Creating a RDF report:	8
Lab 2: XML Publisher Report Using Data Template:	23
Step 1: Create the table and insert records in the table from which you want to print the data.....	23
Step 2: Create the data definition template.....	23
Step 3: Create data definition in xml publisher and attach the data definition template to the data definition	24
Step 4: Create Concurrent Program to generate the Data XML output	26
Step 5: Assignment of concurrent program to responsibility.....	28
Step 6: Execute the concurrent program.....	28
Step 7: Creation of RTF Template	30
Step 8: Registering the RTF Template.....	32
Step 9: Repeat Step 4 again and see the output	33

Getting Started

Overview

This lab book is a guided tour for learning XML Publisher basics. It comprises 'To Do' assignments. Follow the steps provided and work out the 'To Do' assignments.

Setup Checklist for EAS Oracle Application

Here is what is expected on your machine in order for the lab to work.

Minimum System Requirements

- Intel Pentium III or higher
- Microsoft Windows 95, 98, or NT 4.0, 2k, XP, Windows 7
- Memory: 512 MB of RAM (1 GB or more recommended)
- Internet Explorer 6.0 or higher
- Forms and Reports 6i or Higher
- Putty
- Filezilla
- PLSQL Developer
- Oracle Client
- Workflow Builder
- BI Publisher Desktop
- Java 1.6.0_07
- jinitator13118
- Access to Oracle application instance

Instructions

- For all coding standards refer Appendix A. All lab assignments should refer coding standards.
- Create a directory by your name in drive <drive>. In this directory, create a subdirectory XML Publisher>_assgn. For each lab exercise create a directory as lab <lab number>.

XML Publisher Basic

Introduction

Oracle XML Publisher is a template-based publishing solution delivered with the Oracle E-Business Suite. It provides a new approach to report design and publishing by integrating familiar desktop word processing tools with existing E-Business Suite data reporting. XML Publisher leverages standard, well-known technologies and tools, so you can rapidly develop and maintain custom report formats.

The flexibility of XML Publisher is a result of the separation of the presentation of the report from its data structure. The collection of the data is still handled by the E-Business Suite, but now you can design and control how the report outputs will be presented in separate template files. At runtime, XML Publisher merges your designed template files with the report data to create a variety of outputs to meet a variety of business needs, including:

- Customer-ready PDF documents, such as financial statements, marketing materials, contracts, invoices, and purchase orders utilizing colors, images, font styles, headers and footers, and many other formatting and design options.
- HTML output for optimum online viewing.
- Excel output to create a spreadsheet of your report data.
- "Filled-out" third-party provided PDF documents. You can download a PDF document, such as a government form, to use as a template for your report. At runtime, the data and template produce a "filled-out" form.

Flat text files to exchange with business partners for EDI and EFT transmission.

The following graphic displays a few sample documents generated by XML Publisher.

Process Overview

Creating customized reports using XML Publisher can be divided into two phases:
Design Time and Runtime.

Design Time

1. Register the E-Business Suite report as a Data Definition in the Template Manager. Create a Data Definition in the Template Manager for E-Business Suite reports that you wish to customize using XML Publisher.

2. Design your template.

Your template files can be either in Rich Text Format (RTF) or Portable Document Format (PDF). RTF is a specification used by many word processing applications, such as Microsoft Word. You design the template using your desktop word processing application and save the file as an RTF file type (.rtf extension).

Insert basic markup tags to the document to prepare it for merging with the XML data. XML Publisher

Recognizes the formatting features that you apply and converts them to XSL-FO. Use Adobe Acrobat to apply markup tags to your custom-designed or downloaded PDF template.

3. Register your Template in the Template Manager. When you create the template in the Template Manager, you register and upload your RTF or PDF template files. The Template must be assigned to the Data Definition Code of the E-Business Suite report with which it will be merged.

4. Add desired translations of your RTF template. XML Publisher's translation utility allows you to extract the translatable strings from your template into an XLIFF file. Translate the strings in this file and re upload to the Template Manager to make the translation available at runtime.

1-4 Oracle XML Publisher Report Designer's Guide.

Runtime

1. Set the concurrent program to generate XML.

2. Run the concurrent program using standard request submission.

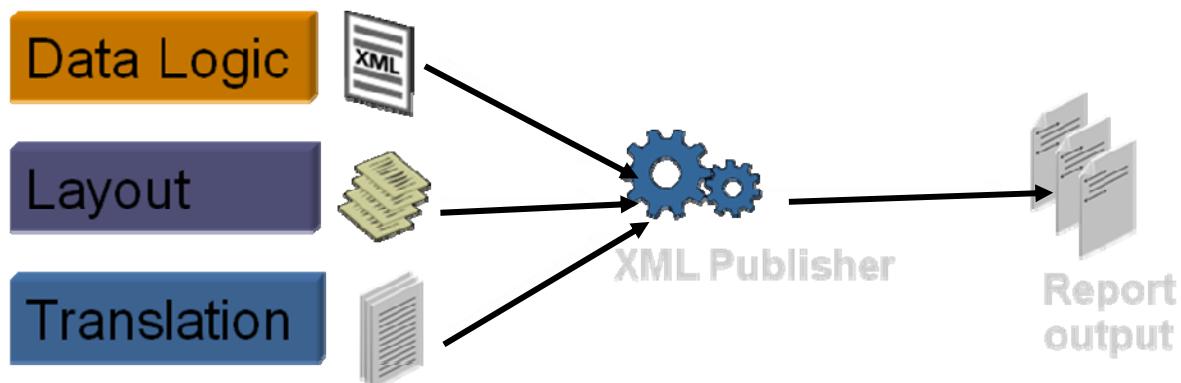
• If you are using the HTML-based Schedule Request interface, select the template and output format from the **Layout** page of the process train.

XML Publisher Concept

XML Publisher breaks the three components apart and treats them separately at design time and at runtime the three are brought back together by XML Publisher to generate the final formatted, translated output.

There is an immediate gain in that this model is far more flexible i.e. a single data definition can support multiple layouts, multiple translations can be applied at runtime to generate translated output. This leads to a reduction in maintenance costs for all concerned.

Separate data / layout / UI translation



- ✓ More flexibility
- ✓ Reduced maintenance

Data Logic: Data extracted from the database and converted into a XML string.

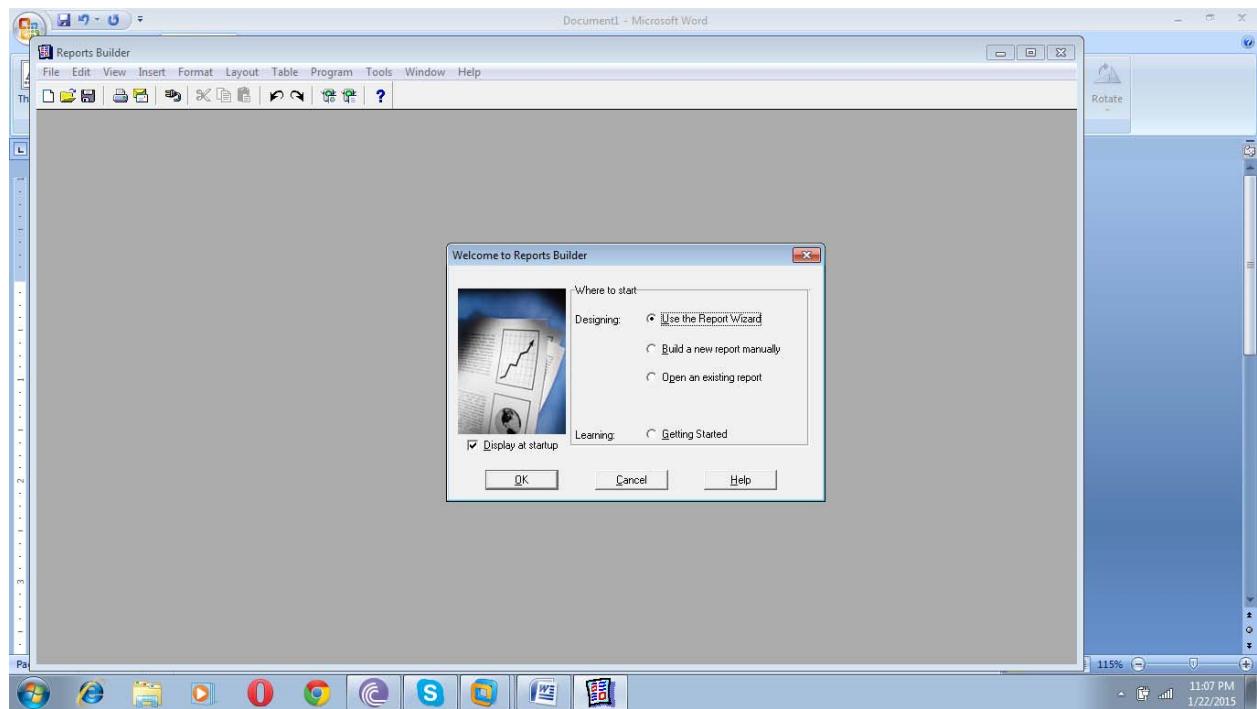
Layout: The layout templates to be used for final output are stored and managed in the Template manager.

Translation: The translation handler will manage the translation that is required at run time.

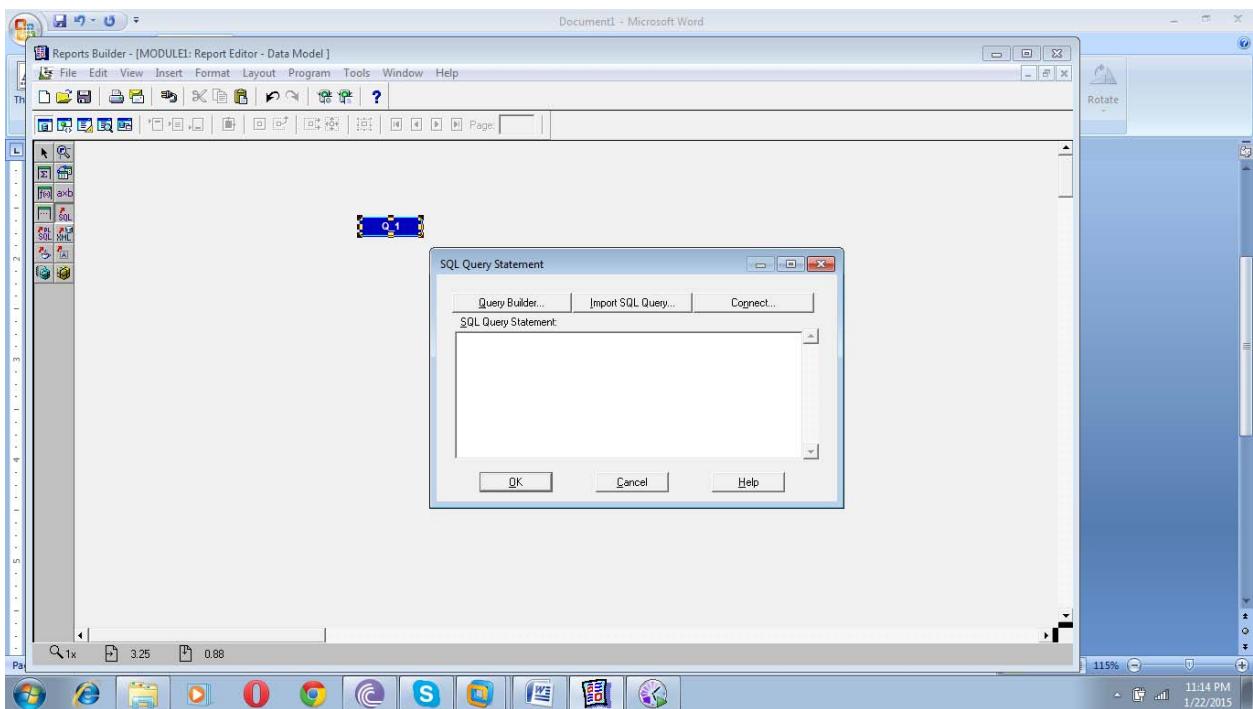
Lab1: Creating a xml publisher report by using rdf report

Creating a RDF report:

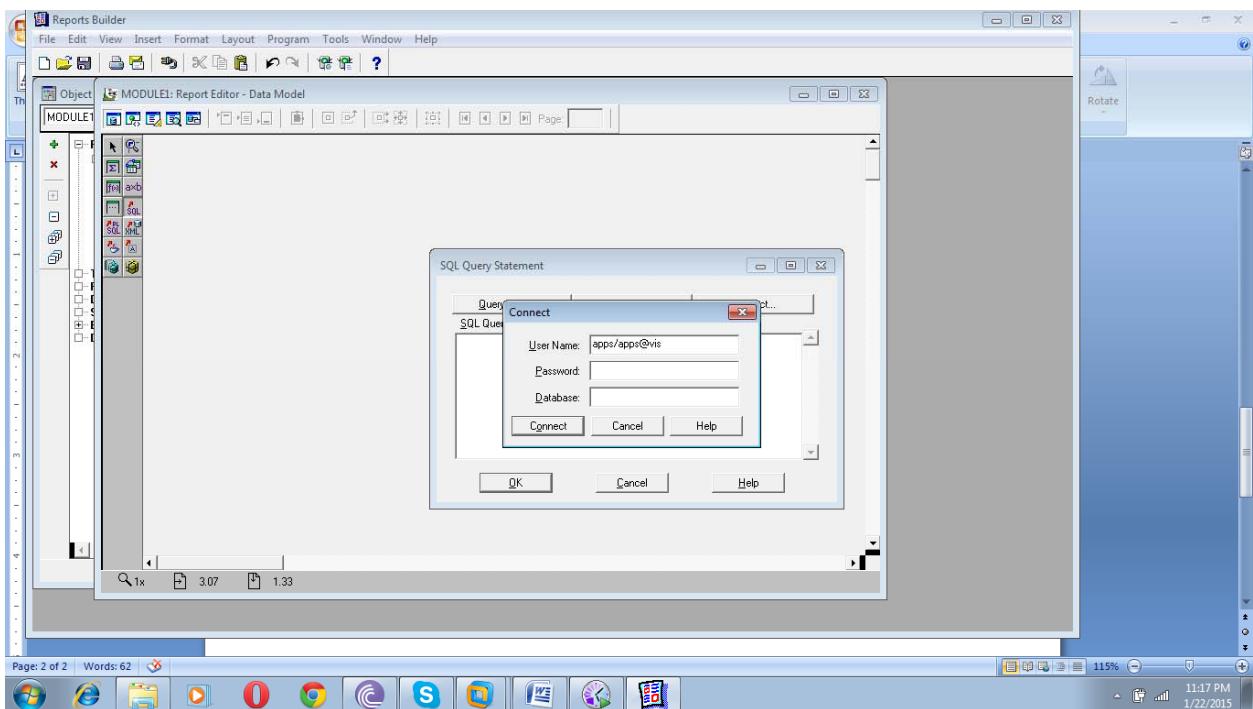
Login to Report Builder tool and select the button build a new report manually.



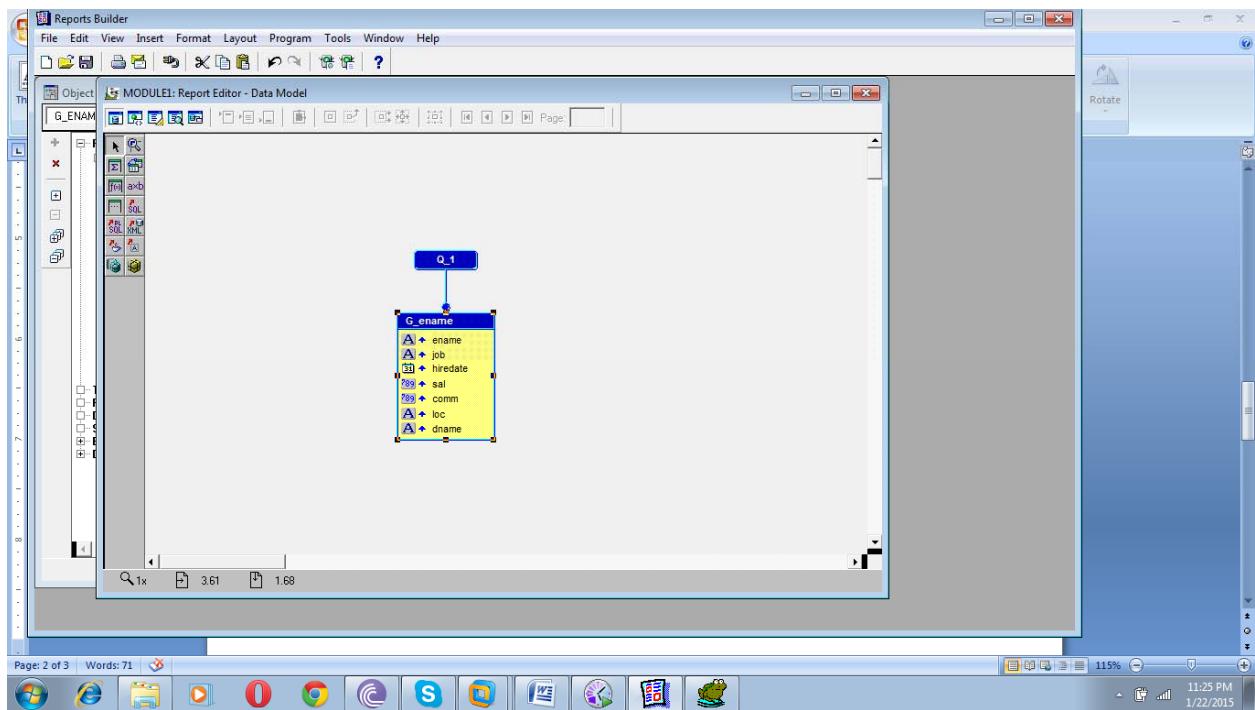
- Now you can see the Module1: Report-editor Data Model. You have to select sql box for write a query to meet the requirement.



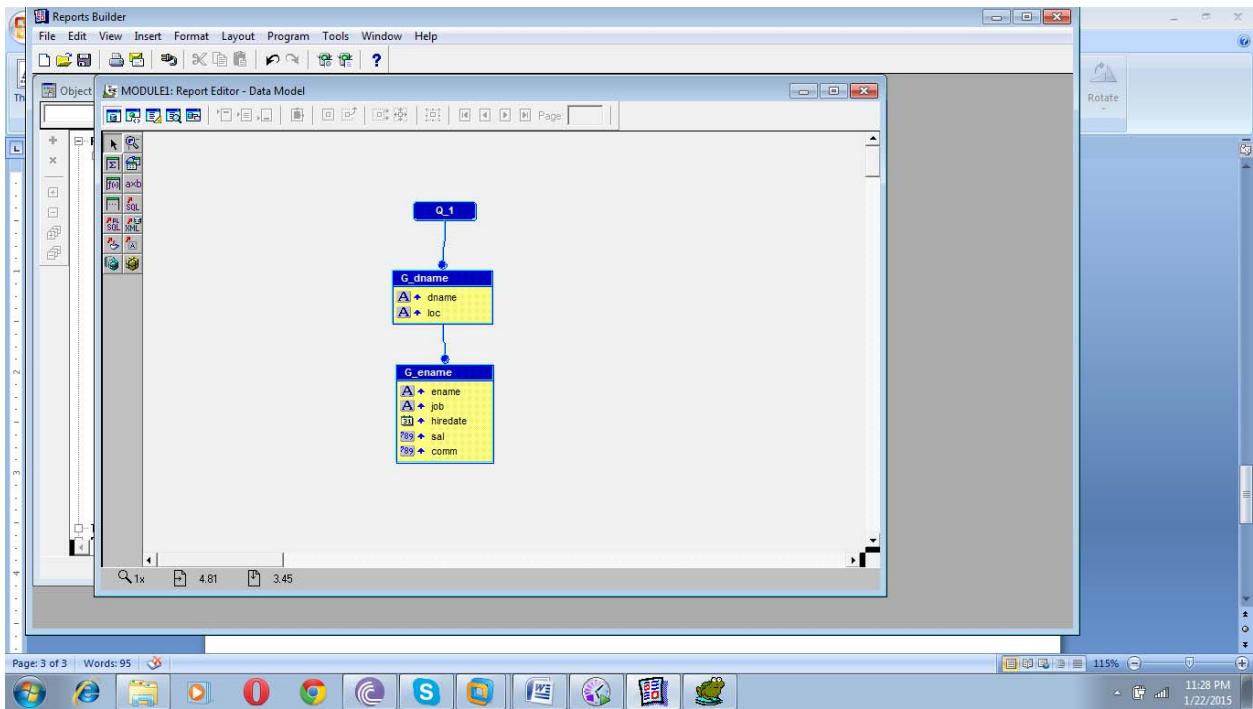
- Now you have select connection button for connection of the database.



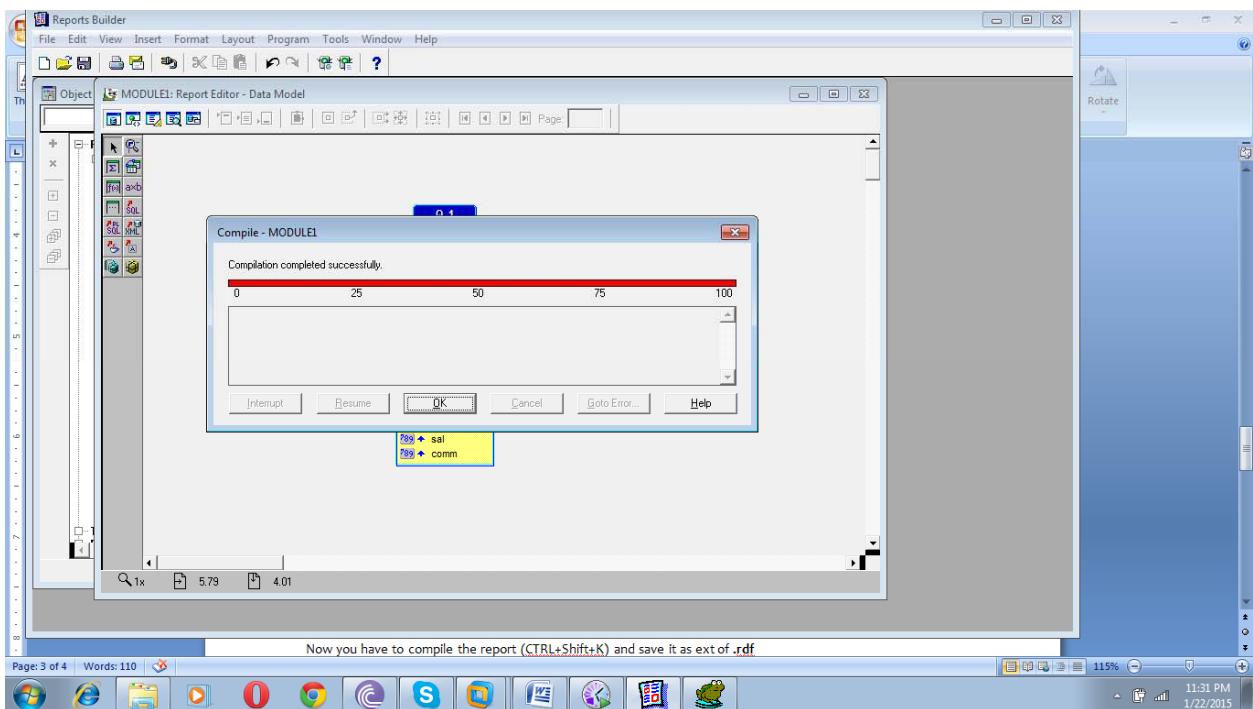
- Then write the query to reach the requirement.



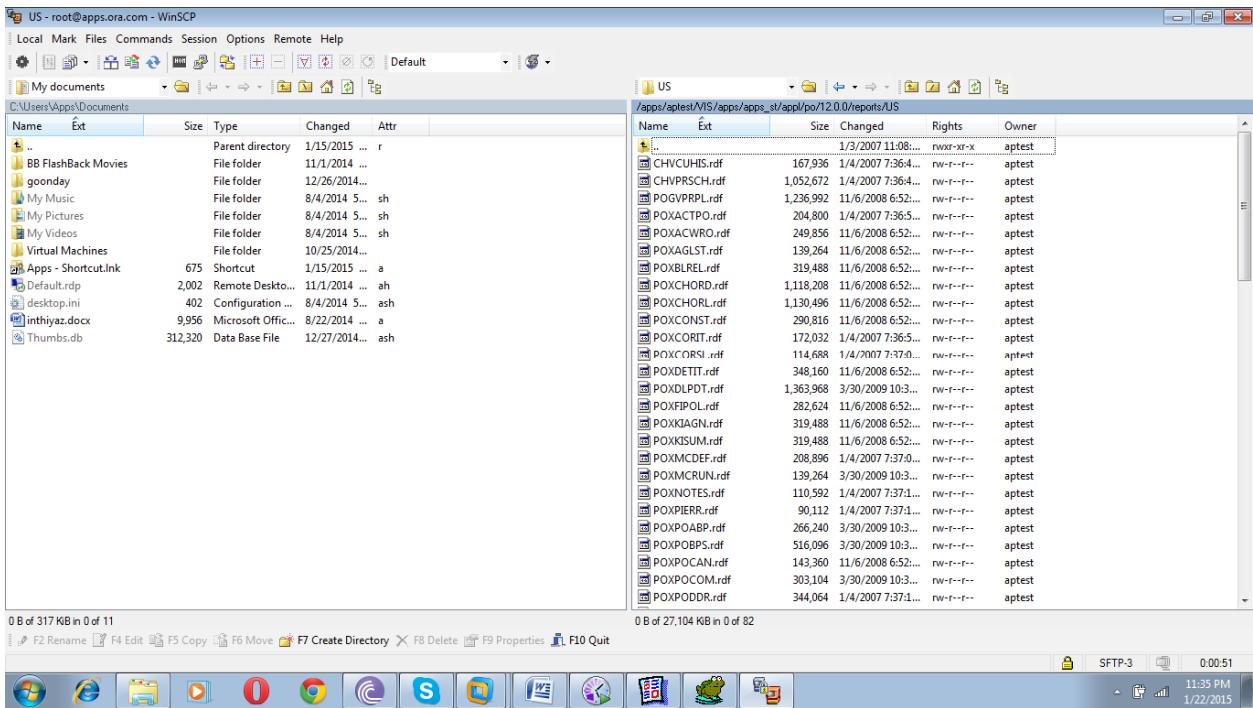
- For meet the requirement purpose you can change the Data Model and you can add Placeholder columns, Summary columns, Formula column, Ref cursor query...



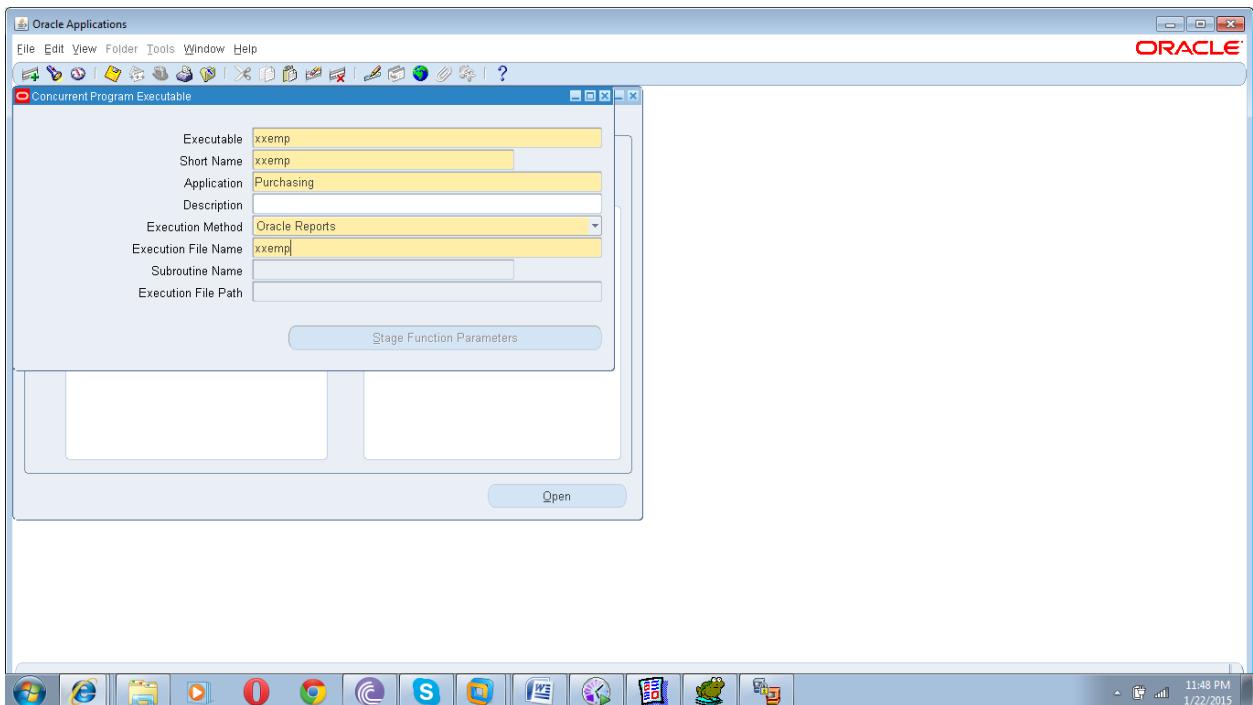
- Now you have to compile the report (CTRL+Shift+K) and save it as ext of .rdf



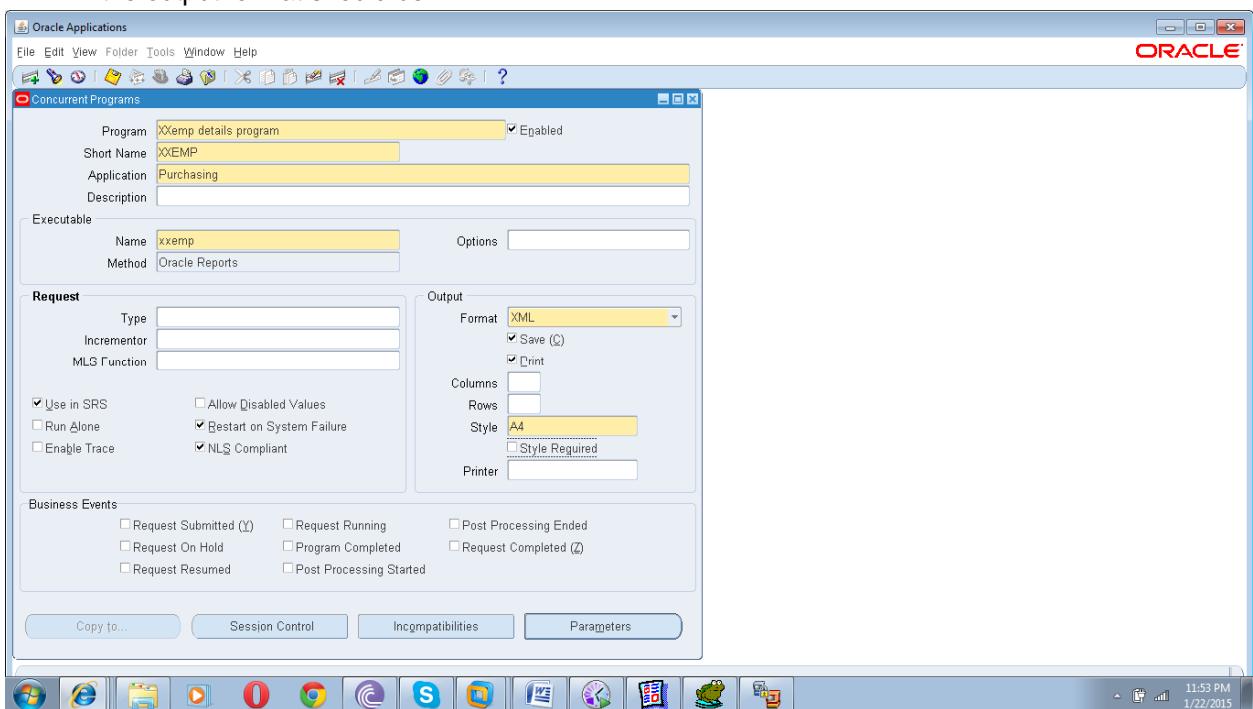
- Now you just move this report to development environment to custom top by using WinSCP



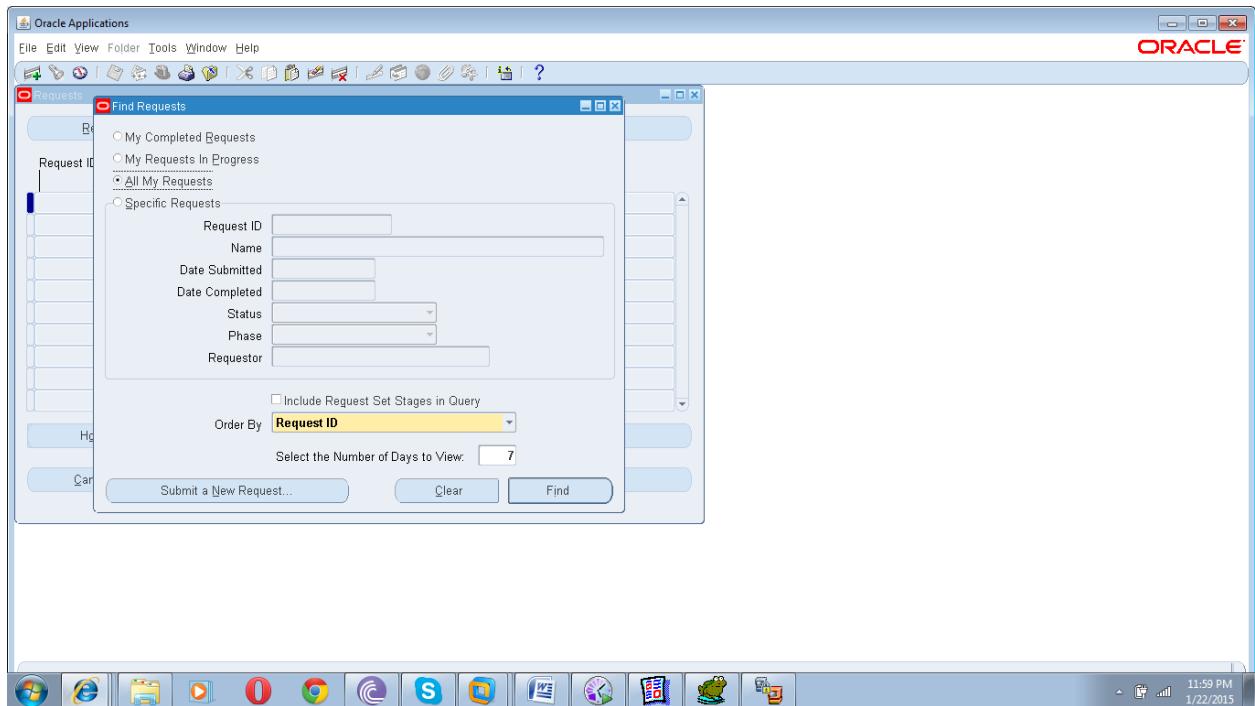
- Now you have to register this rdf report for generation of xml tags.
- You have to login to E-Business suite with user credentials. Now you have to go to the application developer.
- Go to the concurrent program executable window and you have to create executable for the above report.
- Executable method should be Oracle Reports and Executable file name should be the rdf report which ever we have to move the custom top.

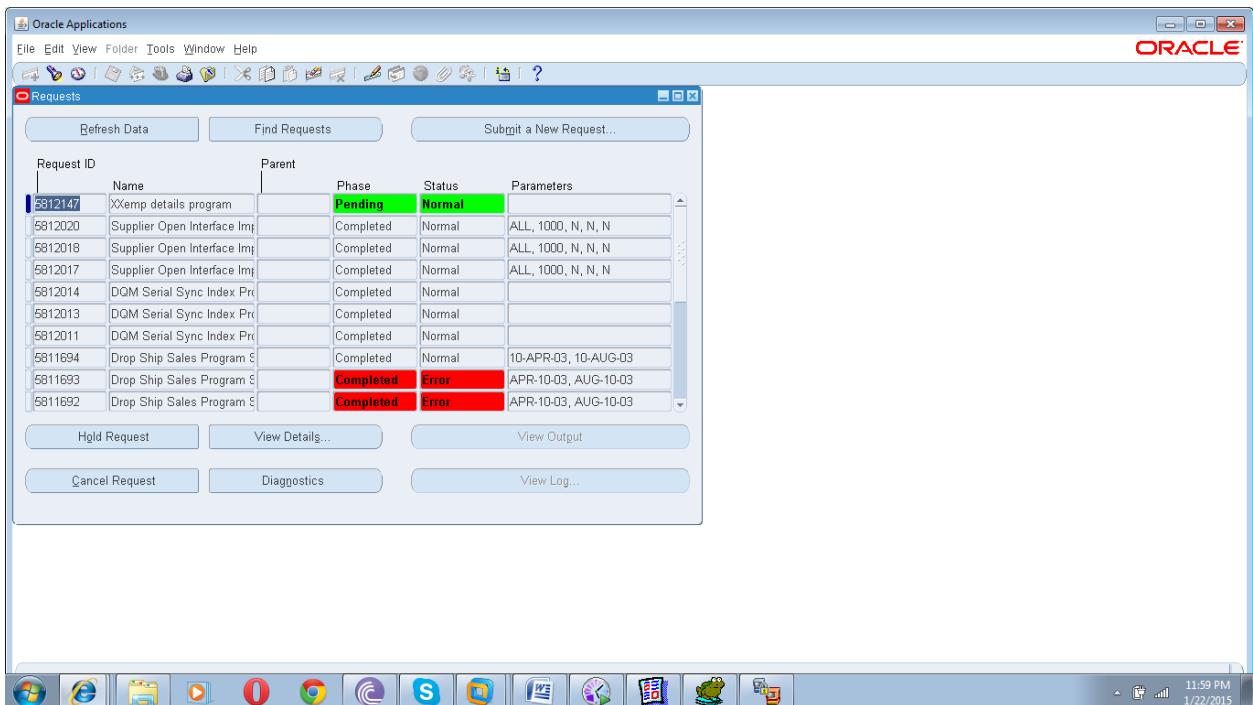


- Now we have to create a concurrent program for this executable. Here the main thing is the output format should be XML.

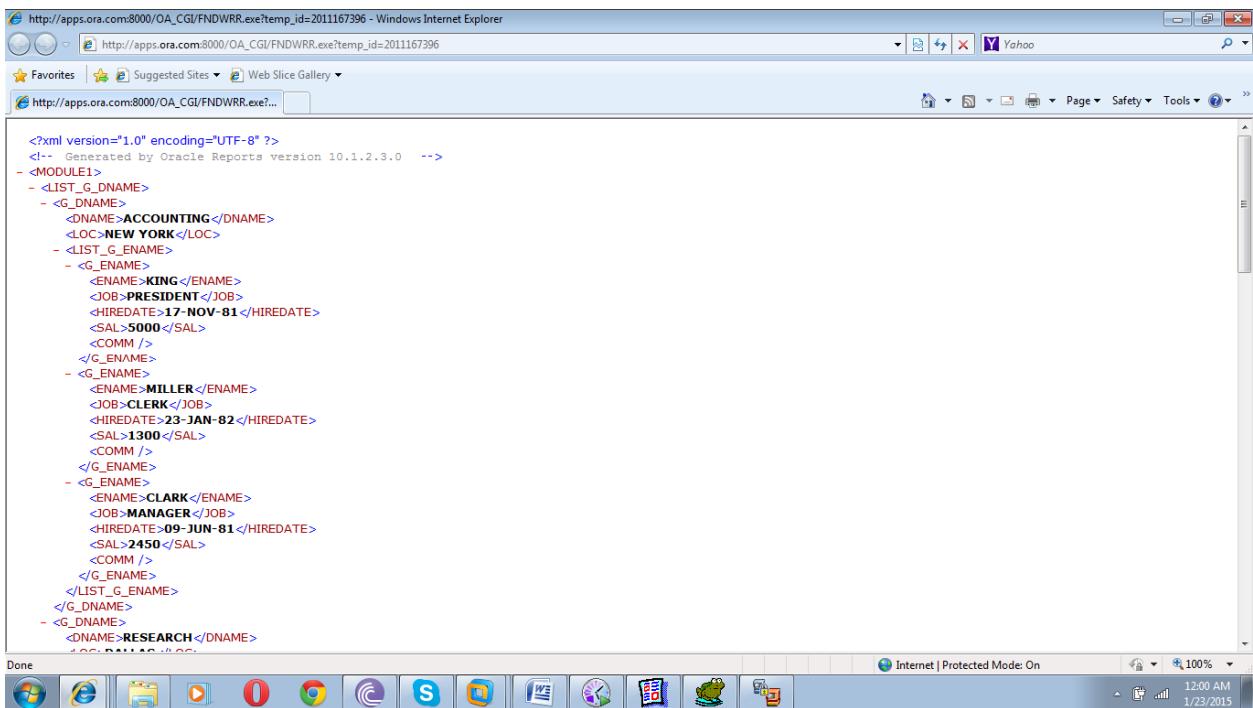


- Now this concurrent program attaches to the request group and the request group should be attach to the particular responsibility.
- We have to submit this program in SRS window and will get Xml tags.





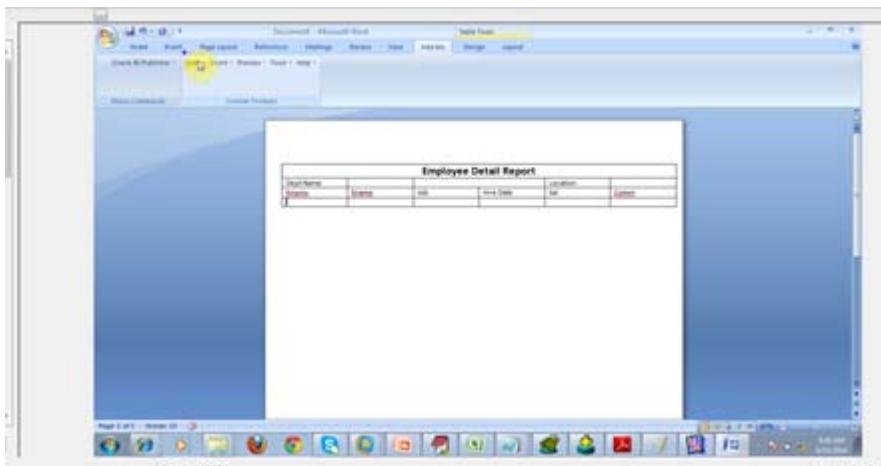
- Phase and status will come as Completed and Normal then xml tags will generate.



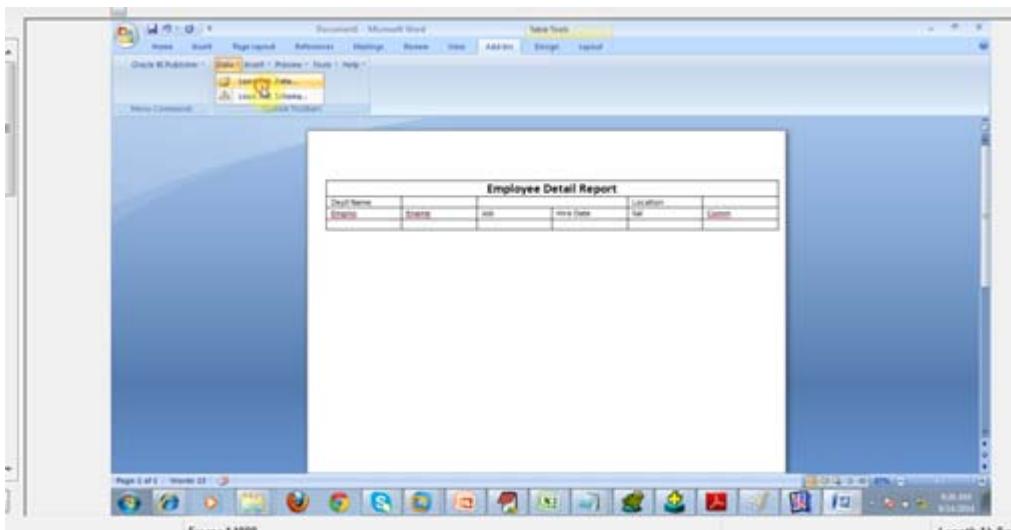
```

<?xml version="1.0" encoding="UTF-8" ?>
<!-- Generated by Oracle Reports version 10.1.2.3.0 -->
<MODULE1>
- <LIST_G_DNAME>
  - <G_DNAME>
    <DNAME>ACCOUNTING</DNAME>
    <LOC>NEW YORK</LOC>
    - <LIST_GENAME>
      - <GENAME>
        <ENAME>KING</ENAME>
        <JOB>PRESIDENT</JOB>
        <HIREDATE>17-NOV-81</HIREDATE>
        <SAL>5000</SAL>
        <COMM />
        <GENAME>
        <ENAME>MILLER</ENAME>
        <JOB>CLERK</JOB>
        <HIREDATE>23-JAN-82</HIREDATE>
        <SAL>1300</SAL>
        <COMM />
        <GENAME>
        <ENAME>CLARK</ENAME>
        <JOB>MANAGER</JOB>
        <HIREDATE>09-JUN-81</HIREDATE>
        <SAL>2450</SAL>
        <COMM />
        <GENAME>
        <LIST_GENAME>
        <G_DNAME>
        <DNAME>RESEARCH</DNAME>
        <LOC>CHICAGO</LOC>
      - <G_DNAME>
    
```

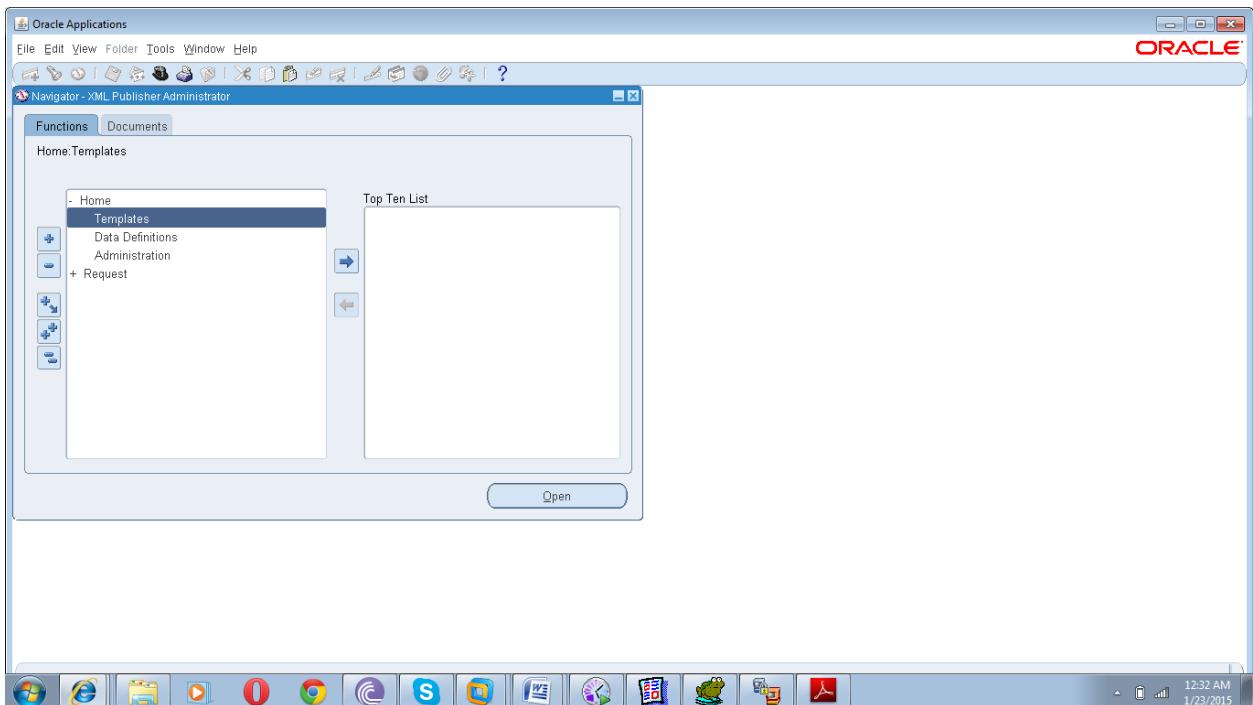
- Now we have to save this xml file with extension of .xml.
- Next we have to Design the Rtf and will load this xml tags to that file.
- Open a word document and design the report as per client requirement.



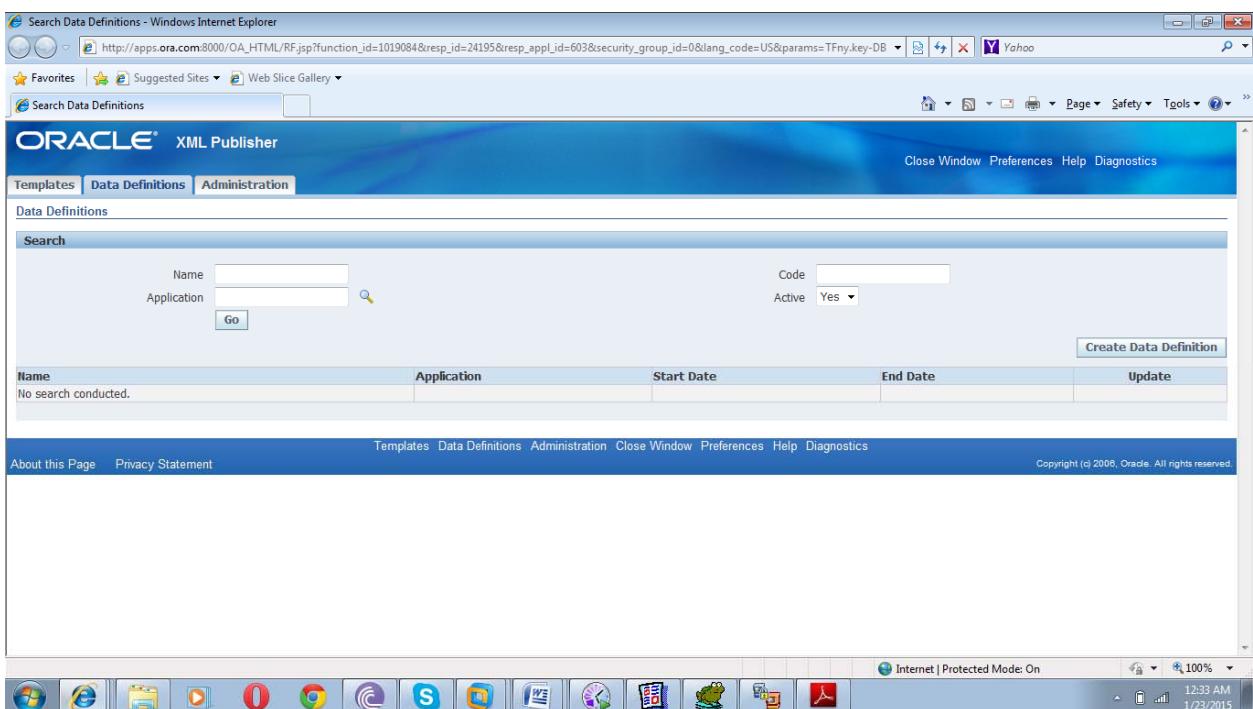
- Now we have to load the xml data and use the business logics to achieve the client requirements.



- Now we have to save this rtf file with extension of Rich Text Format (rtf).
- Now we have to register this rtf file in the xml publisher responsibility.
- Xml Publisher Administrator -> Home - >Data Definitions

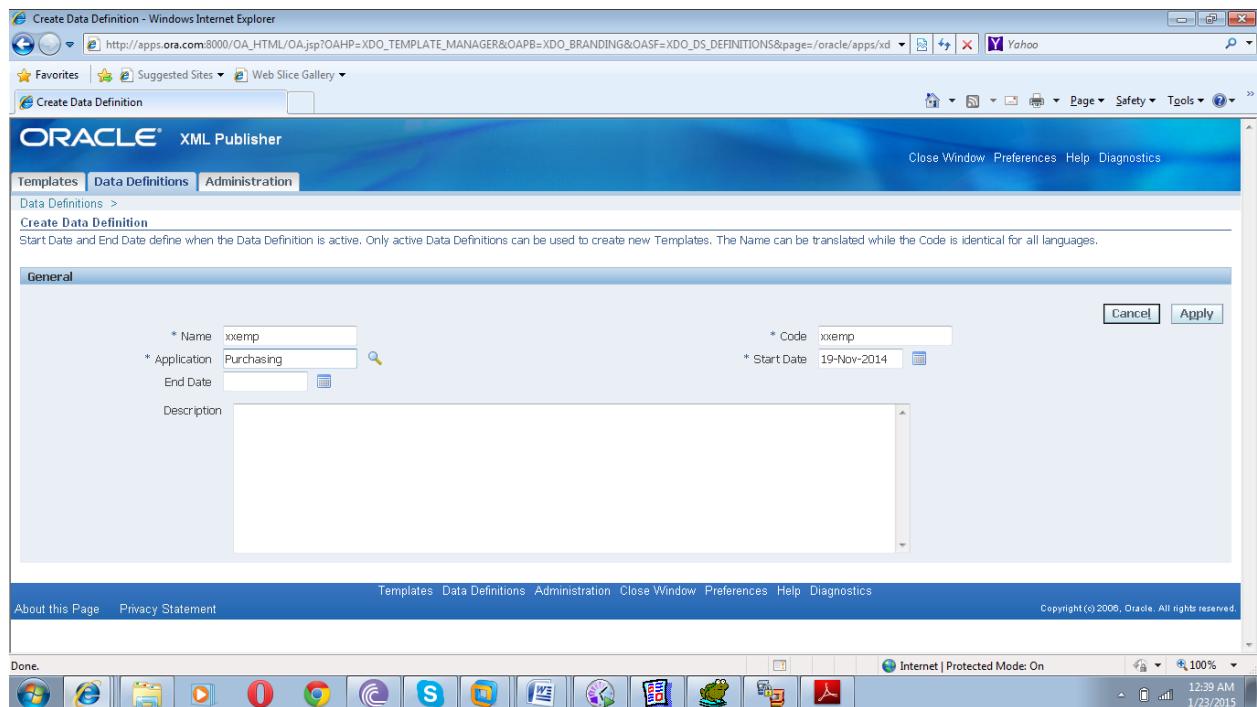


Click on to the Data Definitions you will get xml publisher window

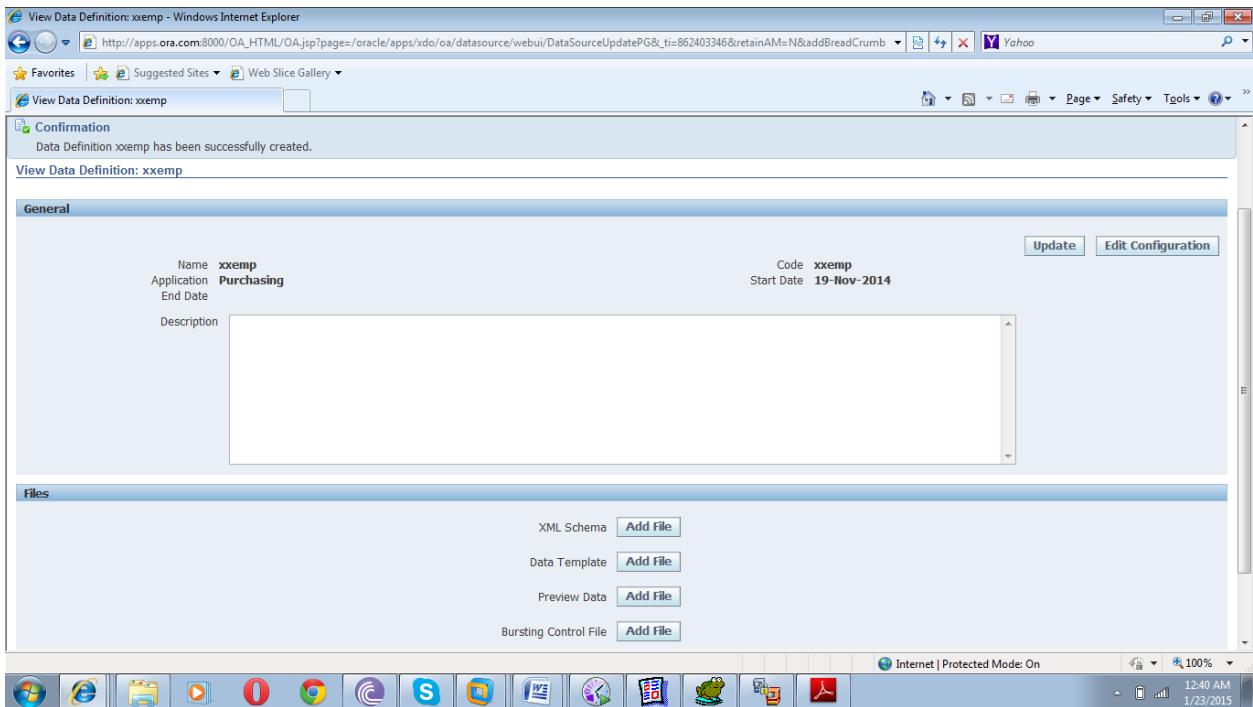


- Click on create data definitions and enter the mandatory fields then press apply

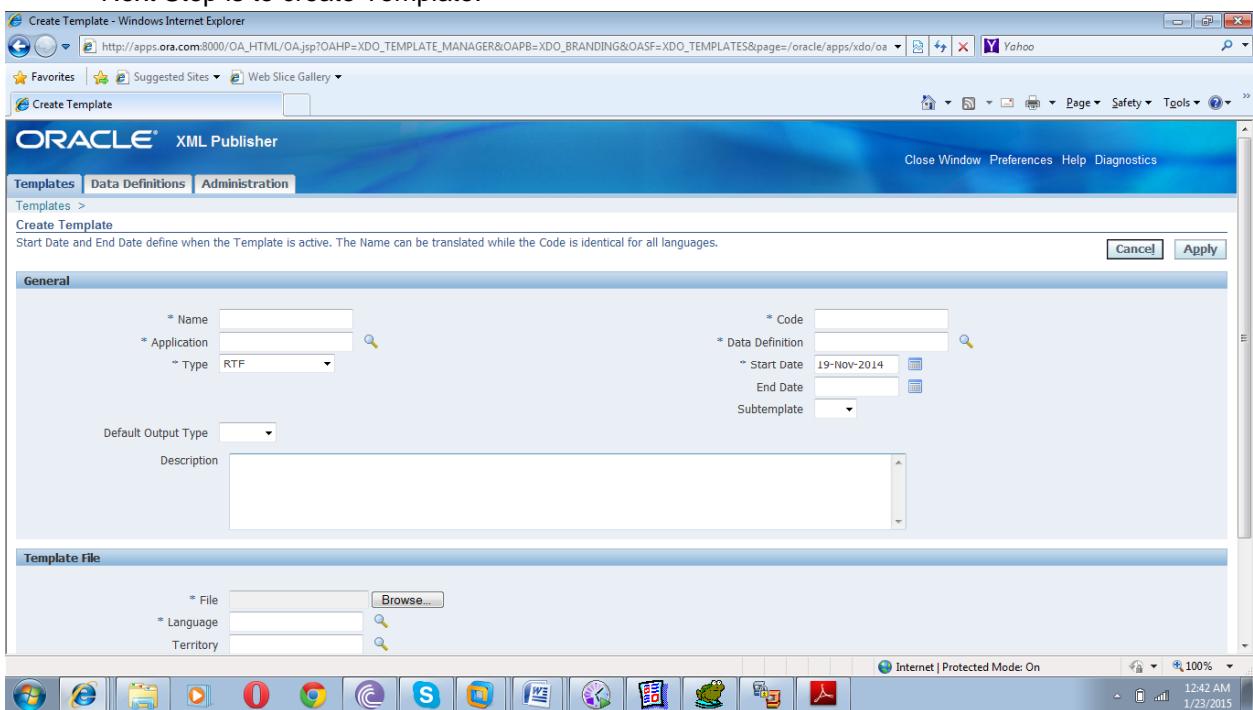
Name: User friendly
Code: Concurrent program short name
Application: APPLICATION of Source rdf (ex: Custom_appl, Purchasing)
Start Date: Developing date



- Data Definition has been successfully created.



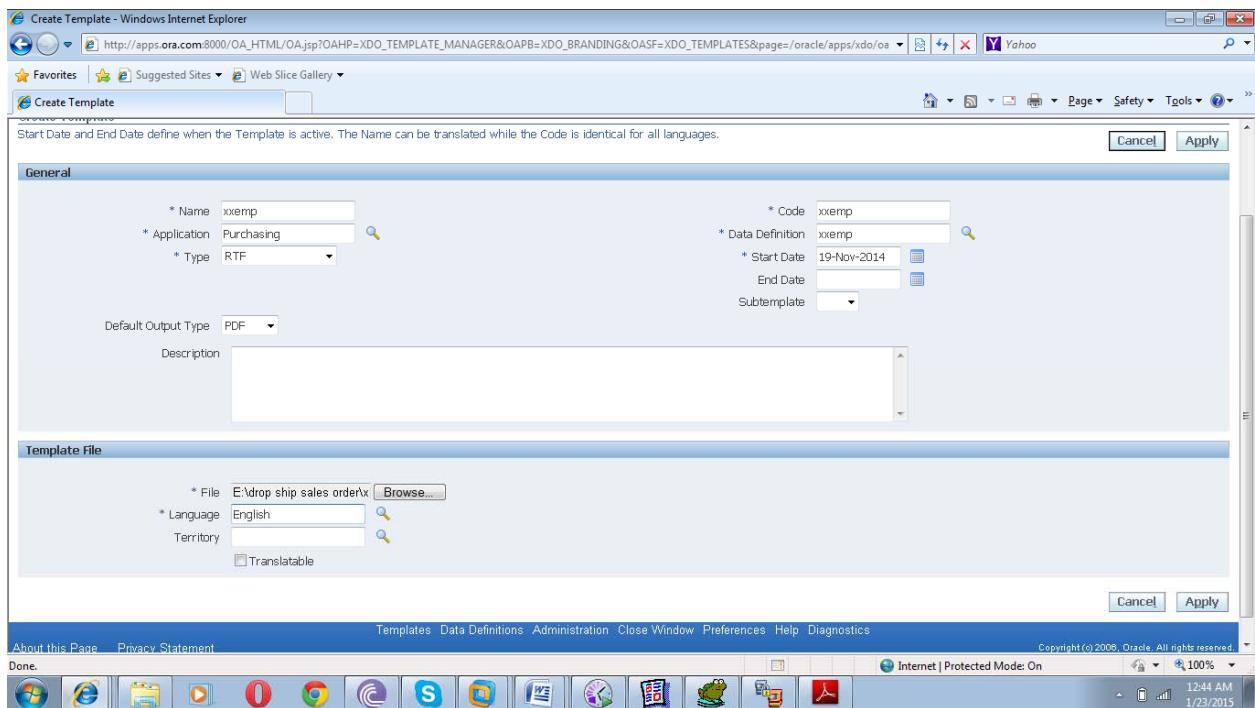
- Next Step is to create Template.



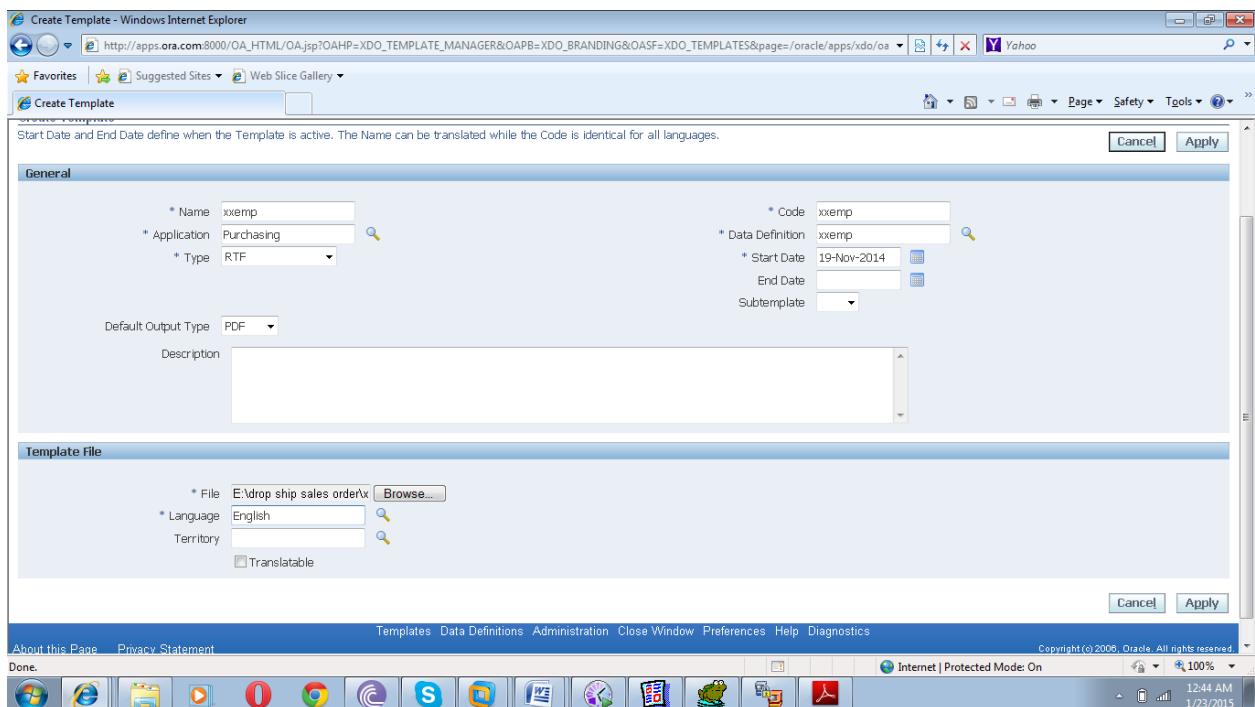
- Enter the mandatory fields as :

Name: User friendly
Application: APPLICATION of Source rdf (ex: Custom_appl, Purchasing)
Code: Concurrent program short name
Data Definition: Created Data Definition
Type: Rtf
Start Date: Developing date
Default Output Type: Provide the Output Type (PDF , Exls..)
File: Rtf File to Browse
Language: Eng

- Browse the rtf file and mention the language then click applies.



- Template has been successfully created



- Now you have to go to the related xml publisher responsibility and submit the concurrent program you will get output as below.

1421953120119out.pdf - Adobe Reader

File Edit View Document Tools Window Help

1 / 1 139% Find

Employee Details Report				
Dept Name	ACCOUNTING		Loc	NEW YORK
EmpName	Sal	Job	Commission	Hiredate
KING	5000	PRESIDENT		17-NOV-81
MILLER	1300	CLERK		23-JAN-82
CLARK	2450	MANAGER		09-JUN-81
Dept Name	RESEARCH		Loc	DALLAS
EmpName	Sal	Job	Commission	Hiredate
JONES	2975	MANAGER		02-APR-81
SCOTT	3000	ANALYST		09-DEC-82
FORD	3000	ANALYST		03-DEC-81
SMITH	800	CLERK		17-DEC-80
ADAMS	1100	CLERK		12-JAN-83
Dept Name	SALES		Loc	CHICAGO
EmpName	Sal	Job	Commission	Hiredate
WARD	1250	SALESMAN	500	22-FEB-81
MARTIN	1250	SALESMAN	1400	28-SEP-81
TURNER	1500	SALESMAN	0	08-SEP-81
JAMES	950	CLERK		03-DEC-81
ALLEN	1600	SALESMAN	300	20-FEB-81
BLAKE	2850	MANAGER		01-MAY-81

12:28 AM
1/23/2015

Lab 2: XML Publisher Report Using Data Template:

Step 1: Create the table and insert records in the table from which you want to print the data

```
CREATE TABLE APPS.EMP
(
  EMPNO NUMBER(4) NOT NULL,
  ENAME VARCHAR2(10),
  JOB VARCHAR2(9),
  MGR NUMBER(4),
  HIREDATE DATE,
  SAL NUMBER(7, 2),
  COMM NUMBER(7, 2),
  DEPTNO NUMBER(2)
);
```

Step 2: Create the data definition template

```
<dataTemplate name="Purchase_Order" description="template description" version="1.1">
<parameters>
<parameter name="dept_no" datatype="number"/>
</parameters>
<dataQuery>
<sqlStatement name="a"><![CDATA[select empno,ename,job,mgr,hiredate,sal,comm,deptno
from emp where deptno=nvl(:dept_no,deptno)]]>
      </sqlStatement>
</dataQuery>

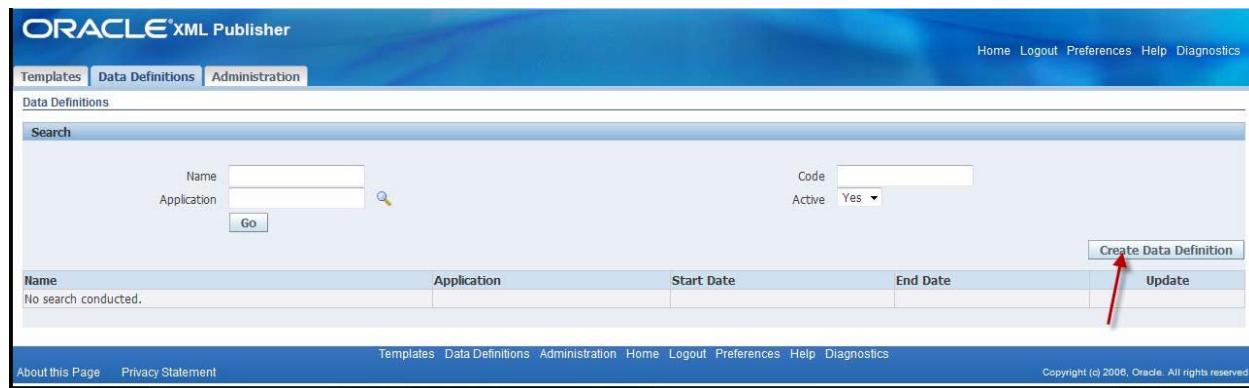
<dataStructure>
<group name="new" source="a">
<element name="Employee_NUMBER" value="empno"/>
<element name="EMPLOYEE_NAME" value="ename"/>
<element name="JOB" value="job"/>
<element name="MGR" value="mgr"/>
<element name="HIREDATE" value="hiredate"/>
<element name="SALARY" value="sal"/>
<element name="COMMISION" value="comm"/>
<element name="DEPT_NO" value="deptno"/>
```

```
</group>
</dataStructure>
</dataTemplate>
```

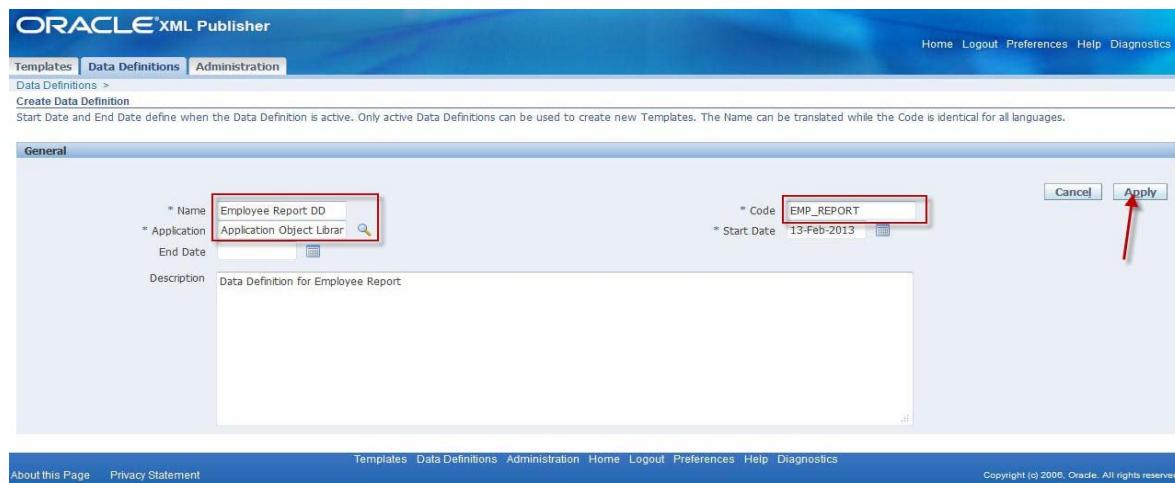
- Save the above file with name of emp.xml.

Step 3: Create data definition in xml publisher and attach the data definition template to the data definition

Navigation: XML Publisher Administrator -> Data Definitions -> Create Data Definition



The screenshot shows the Oracle XML Publisher interface. The top navigation bar includes Home, Logout, Preferences, Help, and Diagnostics. Below it, the main menu has three tabs: Templates, Data Definitions (which is selected), and Administration. Under the Data Definitions tab, there's a sub-menu for Data Definitions and a Search section. The search section contains fields for Name, Application, Code, and Active status (set to Yes). A 'Go' button is next to the search fields. Below the search is a table header with columns: Name, Application, Start Date, End Date, and two buttons: Update and Create Data Definition (which is highlighted with a red arrow). The table body below the header shows a single row with the message "No search conducted." At the bottom of the page, there are links for About this Page, Privacy Statement, and Copyright information.



This screenshot shows the 'Create Data Definition' dialog box. The title bar says 'Create Data Definition'. The General tab is selected. Inside, there are fields for Name ('Employee Report DD'), Application ('Application Object Library'), Code ('EMP_REPORT'), and Start Date ('13-Feb-2013'). There are also 'Cancel' and 'Apply' buttons at the bottom right, with 'Apply' being highlighted by a red arrow. The dialog box also contains a 'Description' field with the text 'Data Definition for Employee Report'. The footer of the dialog box includes links for About this Page and Privacy Statement, and a copyright notice.

ORACLE® XML Publisher

Templates | Data Definitions | Administration

Data Definitions >

Confirmation
Data Definition Employee Report DD has been successfully created.

View Data Definition: Employee Report DD

General

Name	Employee Report DD	Code	EMP_REPORT
Application	Application Object Library	Start Date	13-Feb-2013
End Date			
Description	Data Definition for Employee Report		

Files

XML Schema	Add File
Data Template	Add File
Preview Data	Add File
Bursting Control File	Add File

[Return to Data Definitions](#)

Templates | Data Definitions | Administration | Home | Logout | Preferences | Help | Diagnostics

About this Page | Privacy Statement

Copyright (c) 2006, Oracle. All rights reserved.

ORACLE® XML Publisher

Templates | Data Definitions | Administration

Data Definitions > View Data Definition: Employee Report DD >

Add File for Data Definition: Employee Report DD

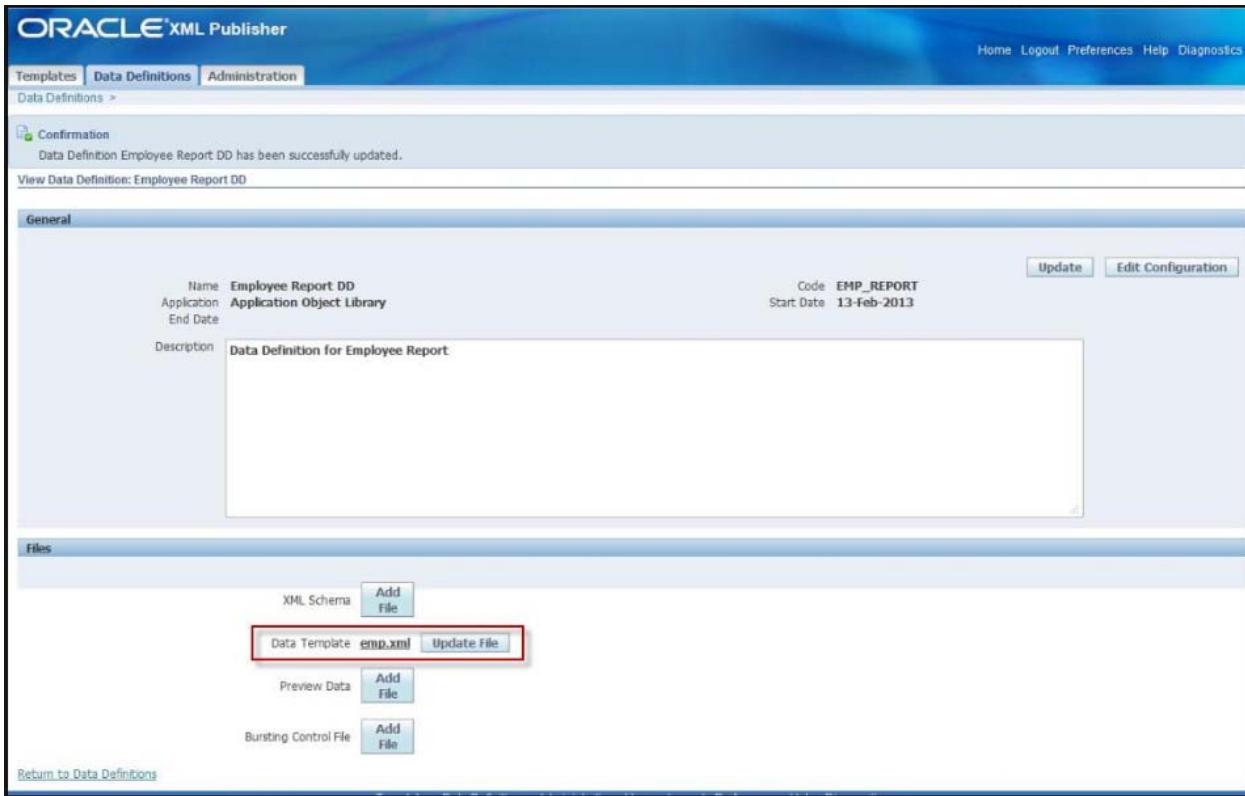
* Data Template C:\Users\shverma\Desktop\

[Browse...](#)

Templates | Data Definitions | Administration | Home | Logout | Preferences | Help | Diagnostics

About this Page | Privacy Statement

Copyright (c) 2006, Oracle. All rights reserved.



The screenshot shows the Oracle XML Publisher interface for managing Data Definitions. The main title bar reads "ORACLE XML Publisher". The navigation menu at the top includes "Templates", "Data Definitions" (which is selected), and "Administration". The sub-menu under "Data Definitions" is "Data Definitions >". A confirmation message is displayed: "Confirmation: Data Definition Employee Report DD has been successfully updated." Below this, a link "View Data Definition: Employee Report DD" is shown. The main content area is titled "General". It contains fields for "Name" (Employee Report DD), "Application" (Application Object Library), "End Date" (not specified), "Code" (EMP_REPORT), "Start Date" (13-Feb-2013), and a "Description" box containing "Data Definition for Employee Report". At the top right of this section are "Update" and "Edit Configuration" buttons. Below the General tab is a "Files" tab. Under "Files", there are sections for "XML Schema" (with an "Add File" button), "Data Template" (listing "emp.xml" with an "Update File" button highlighted by a red box), "Preview Data" (with an "Add File" button), and "Bursting Control File" (with an "Add File" button). At the bottom left of the main content area is a "Return to Data Definitions" link.

Step 4: Create Concurrent Program to generate the Data XML output.

Navigation: System Administrator-> Concurrent-> Program->Define

XDODTEXE is java concurrent program used as a BI Publisher Data template Executable. The main purpose of this executable is to identify data template file (.xml) and execute the data template to generate the raw (output) xml data, which is later used by BI Publisher formatting engine to format as per the layout (RTF, PDF, etc.).

This executable will be used by all the BI Publisher reports (Concurrent Program) which are using Data Template to generate the xml data.

Concurrent Programs

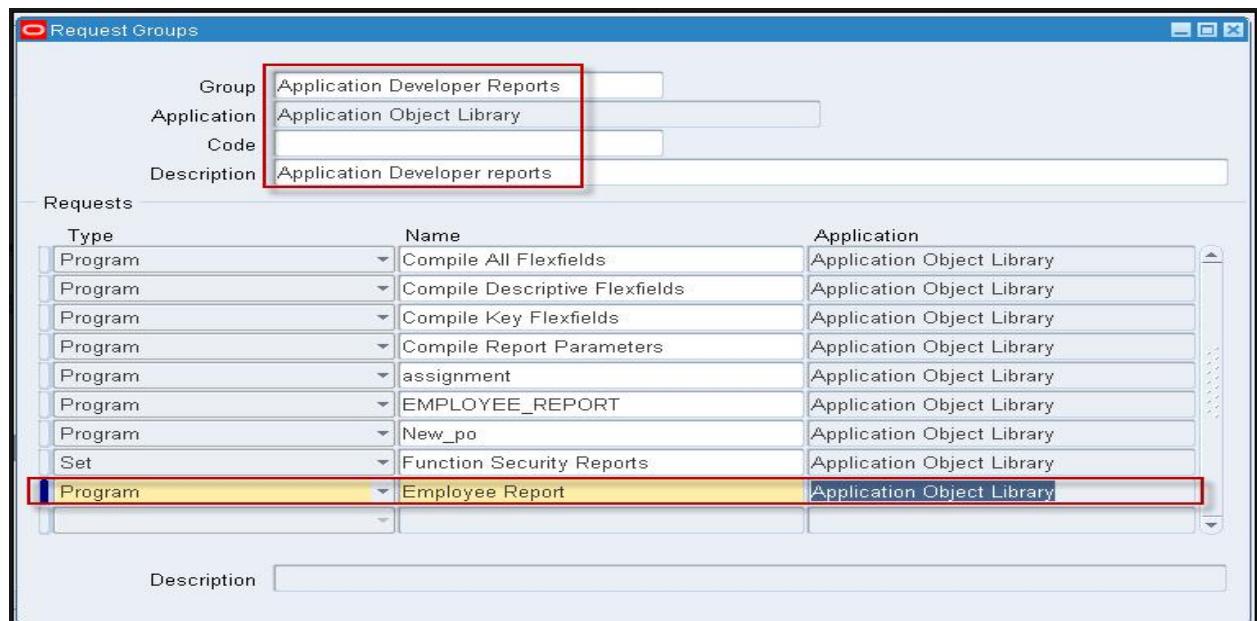
Program	Employee Report	<input checked="" type="checkbox"/> Enabled
Short Name	EMP_REPORT	
Application	Application Object Library	
Description		
Executable	Name: XDODTEXE Method: Java Concurrent Program	Options: <input type="text"/> Priority: <input type="checkbox"/>
Request	Type: <input type="text"/> Incrementor: <input type="text"/> MLS Function: <input type="text"/>	
<input checked="" type="checkbox"/> Use in SRS <input type="checkbox"/> Allow Disabled Values <input type="checkbox"/> Run Alone <input checked="" type="checkbox"/> Restart on System Failure <input type="checkbox"/> Enable Trace <input checked="" type="checkbox"/> NLS Compliant		
Business Events	<input type="checkbox"/> Request Submitted (Y) <input type="checkbox"/> Request Running <input type="checkbox"/> Request On Hold <input type="checkbox"/> Program Completed <input type="checkbox"/> Request Resumed <input type="checkbox"/> Post Processing Started <input type="checkbox"/> Post Processing Ended <input type="checkbox"/> Request Completed (Z)	
<input type="button" value="Copy to..."/> <input type="button" value="Session Control"/> <input type="button" value="Incompatibilities"/> <input style="background-color: #0070C0; color: white; font-weight: bold; border: 1px solid #0070C0; padding: 2px 10px; border-radius: 10px;" type="button" value="Parameters"/>		

Concurrent Program Parameters

Program	Employee Report		
Application	Application Object Library		
Conflicts Domain	<input type="text"/>		
Security Group	<input type="text"/>		
Seq	Parameter	Description	Enabled
1	Parameter1		<input checked="" type="checkbox"/>
<input type="checkbox"/> Required <input type="checkbox"/> Enable Security			
Validation Value Set: FND_NUMBER Default Type: <input type="text"/>		Description: Positive Integer Number Default Value: <input type="text"/> Range: <input type="text"/>	
<input checked="" type="checkbox"/> Display Display Size: 15 Description Size: 50 Concatenated Description Size: 25 Prompt: Department Number Token: dept_no			

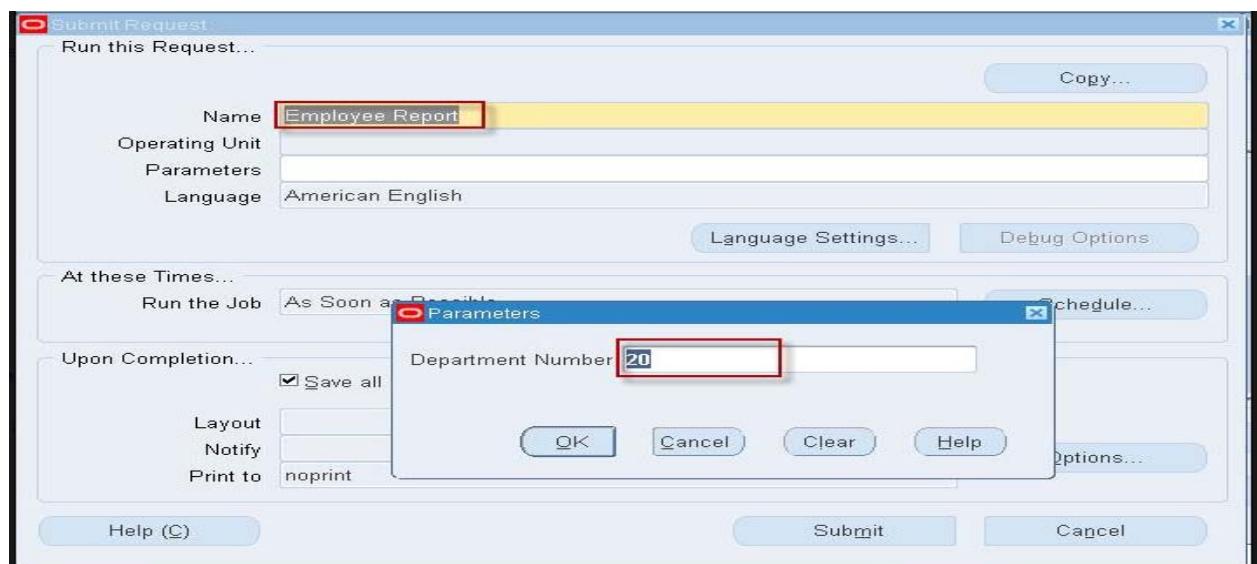
Step 5: Assignment of concurrent program to responsibility

Navigation: System Administrator-> Security-> Responsibility->Request



Step 6: Execute the concurrent program

Navigation: Application Developer-> Concurrent-> Submit a New Request->Single Request



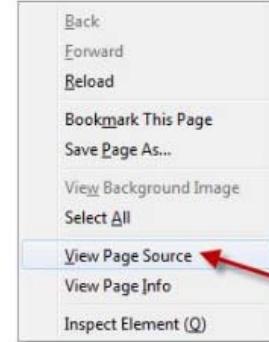


Output will come like this .click on page source and save the page as **EMP_Data.xml**

```

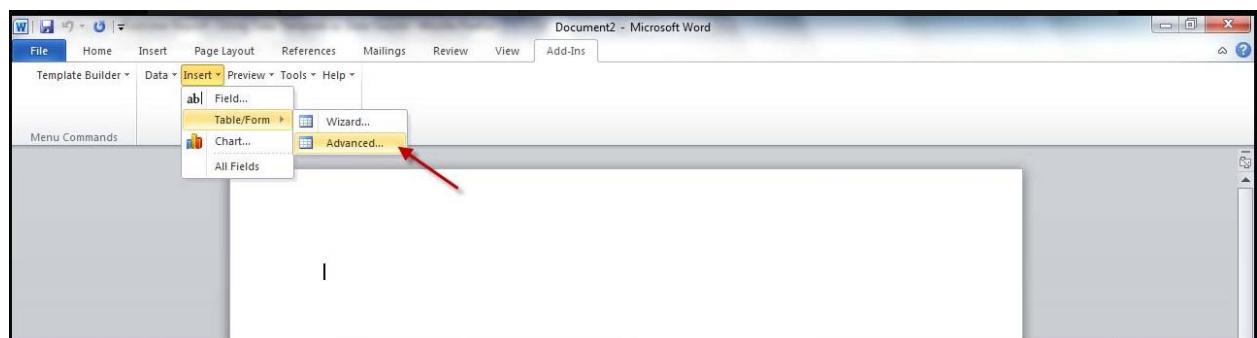
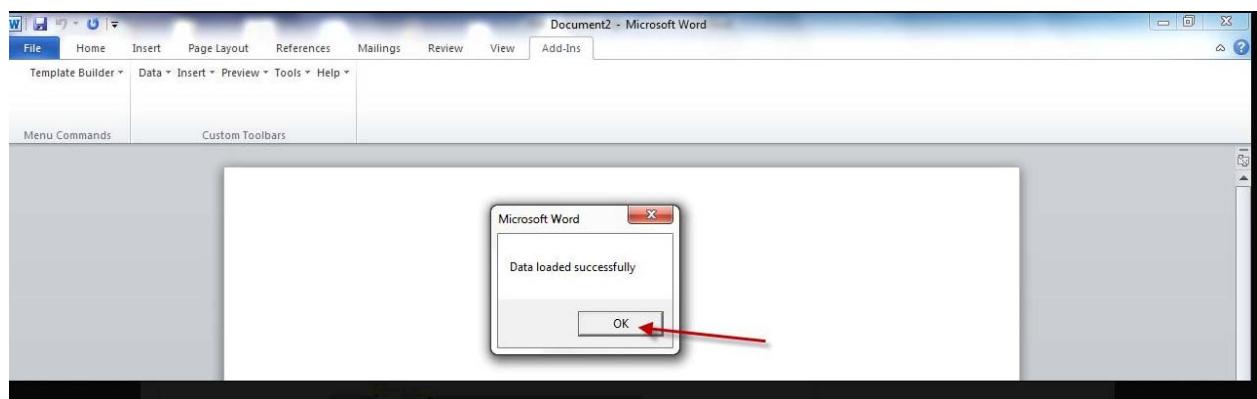
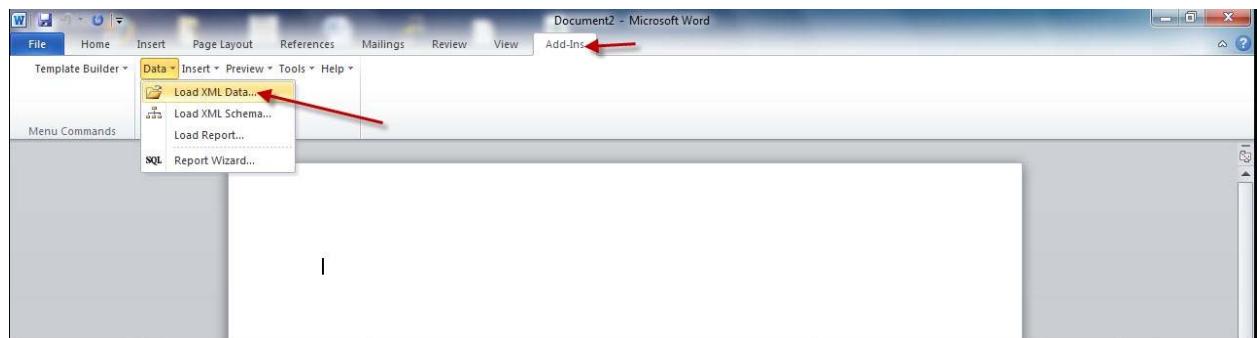
- <PURCHASE_ORDER>
  <DEP/>
- <LIST_NEW>
  - <NEW>
    <EMPLOYEE_NUMBER>7369</EMPLOYEE_NUMBER>
    <EMPLOYEE_NAME>SMITH</EMPLOYEE_NAME>
    <JOB>CLERK</JOB>
    <MGR>7902</MGR>
    <HIREDATE>1980-12-17T00:00:00.000+05:30</HIREDATE>
    <SALARY>800</SALARY>
    <COMMISION>
    <DEPT_NO>20</DEPT_NO>
  </NEW>
  - <NEW>
    <EMPLOYEE_NUMBER>7499</EMPLOYEE_NUMBER>
    <EMPLOYEE_NAME>ALLEN</EMPLOYEE_NAME>
    <JOB>SALESMAN</JOB>
    <MGR>7698</MGR>
    <HIREDATE>1981-02-20T00:00:00.000+05:30</HIREDATE>
    <SALARY>1600</SALARY>
    <COMMISION>300</COMMISION>
    <DEPT_NO>30</DEPT_NO>
  </NEW>
  - <NEW>
    <EMPLOYEE_NUMBER>7521</EMPLOYEE_NUMBER>
    <EMPLOYEE_NAME>WARD</EMPLOYEE_NAME>
    <JOB>SALESMAN</JOB>
    <MGR>7698</MGR>
    <HIREDATE>1981-02-22T00:00:00.000+05:30</HIREDATE>
    <SALARY>1250</SALARY>
    <COMMISION>500</COMMISION>
    <DEPT_NO>30</DEPT_NO>
  </NEW>

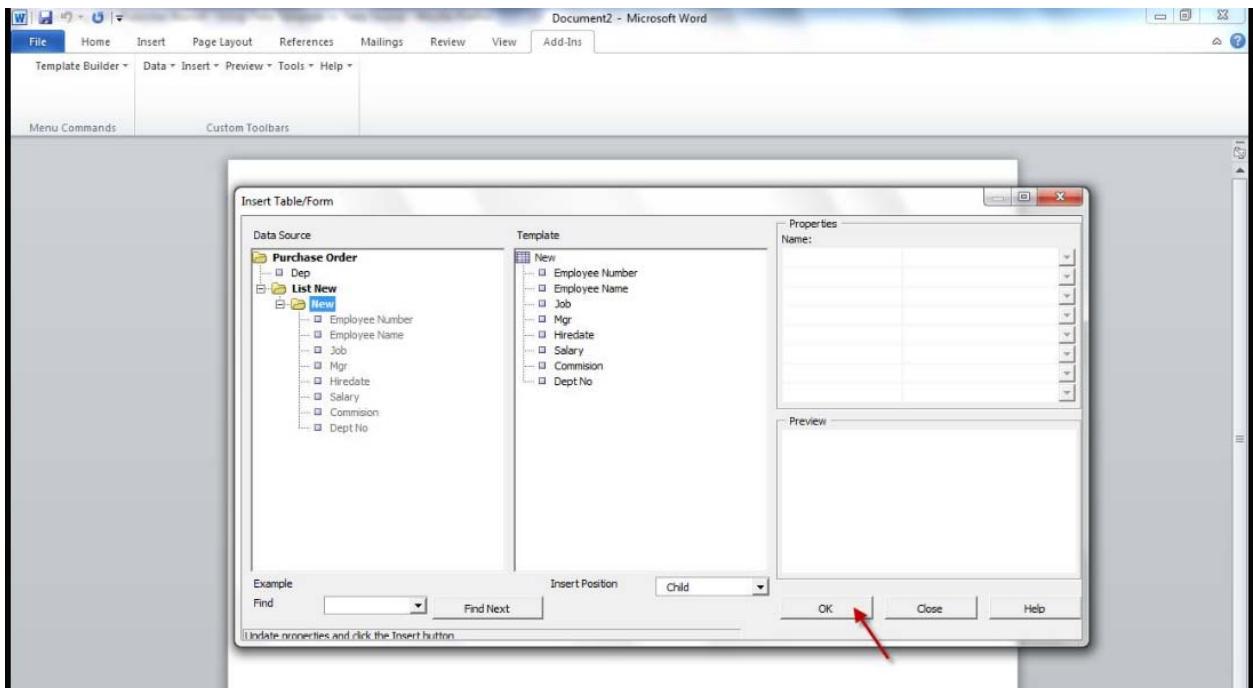
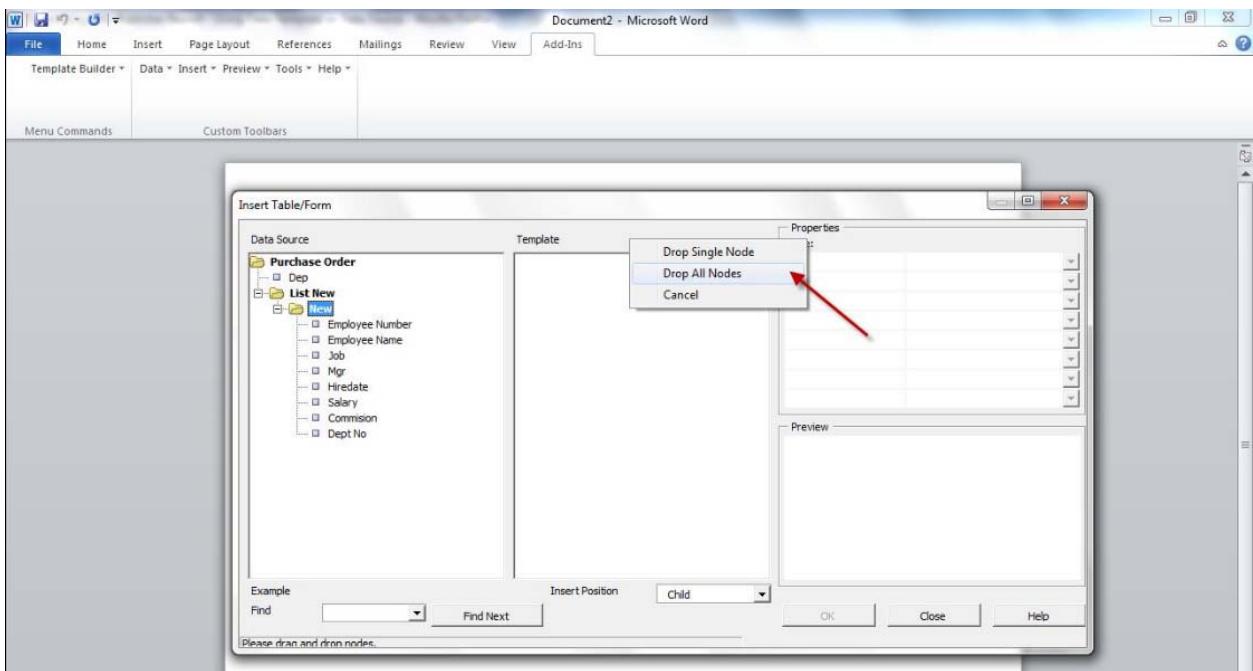
```



Step 7: Creation of RTF Template

- Prerequisite:** You have to install XMLP Template Builder
- Open Microsoft-word file.
 - Click on Add-Ins and then Load XML Data. Give the path of **EMP_Data.xml** file.





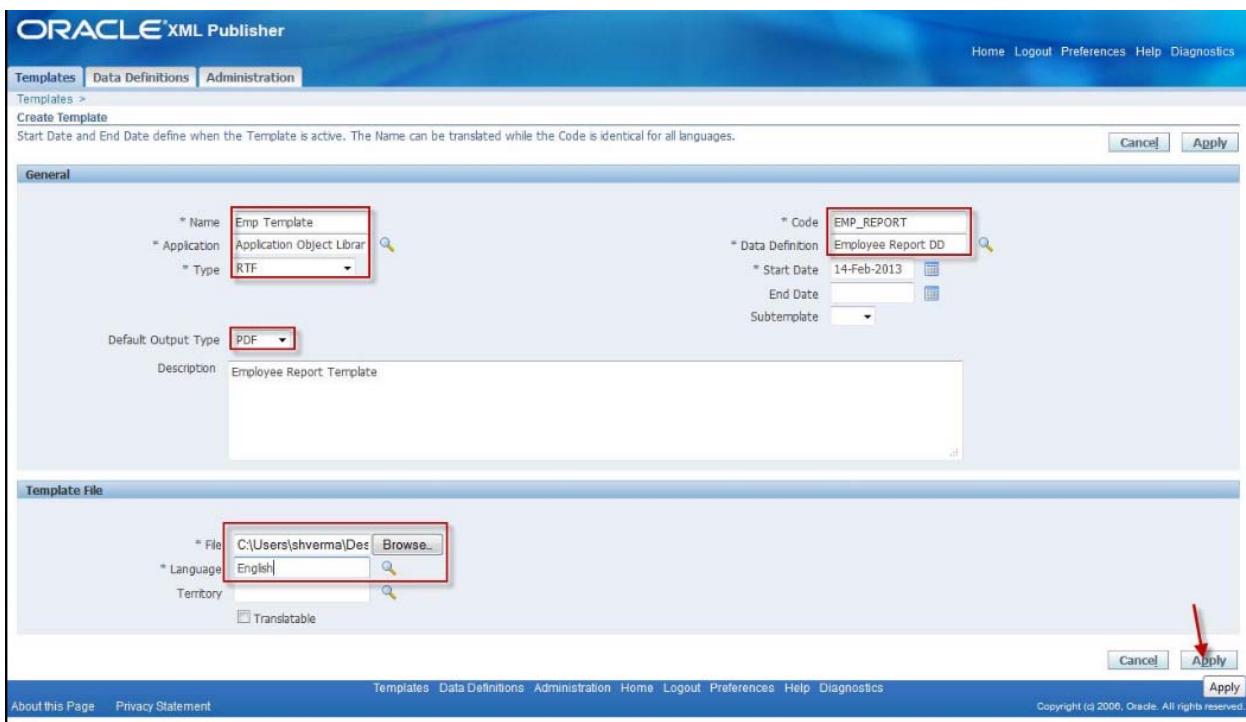


Save the file as **EMP_Template.rtf**

Note: The designing of the report we are doing in this section.

Step 8: Registering the RTF Template

Navigation: XML Publisher Administrator -> Template -> Create Template



Note: The output format of the report we can change in **default output type** section. It may be in excel, html, pdf, rtf.

Step 9: Repeat Step 4 again and see the output

Employee Number	Employee Name	Job	Mgr	Hiredate	Salary	Commission	Dept No
7369	SMITH	CLERK	7902	1980-12-17T00:00:00.000+05:30	800		20
7499	ALLEN	SALESMAN	7698	1981-02-20T00:00:00.000+05:30	1600	300	30
7521	WARD	SALESMAN	7698	1981-02-22T00:00:00.000+05:30	1250	500	30
7566	JONES	MANGER	7839	1981-04-02T00:00:00.000+05:30	2975		20