Dököll Solutions, Inc.

IBM | Oracle

Domino Designer, JDeveloper, DB2, WebLogic, JSF, Xpages, Web Services

Part Two - A

System Requirements

Software Used	Language Used	Protocol/Framework Used
Microsoft Windows 7, 8, 10	VBScript, Batch	Active Directory
Microsoft Internet Explorer	N/A	TCPIP, HTTP
Google Chrome	N/A	TCPIP, HTTP
Mozilla FireFox	N/A	TCPIP, HTTP
Oracle JDeveloper Version 12..x IDE	CSS, HTML, JavaScript, Java	HTTP, TCPIP, Java Server Faces
Oracle Integrated WebLogic Server 12.x	WSDL	HTTP, TCPIP
IBM DB2 Express-C v9.7.4	SQL	SQL Database Server
IBM Notes Domino Designer 8.5.3	CSS, HTML, JavaScript, Java, JavaAgent, XML	HTTP, TCPIP, Xpages

Disclaimer:
Information contained in the following is presented as is. This tutorial assumes you have basic programming and software configuration knowledge. All tutorials are based on Oracle Fusion Middleware and IBM Lotus Notes products, including and not limited to items stated in System Requirements. Should you need to familiarize yourself with JDeveloper or Domino Designer environments, prior to continuing this tutorial, please stop now, and see our Tutorials page: www.dokollsolutionsinc.com or consult your favourite or preferred Search Engine/Company sites for support...

Introduction:

Now that you have Created New App with JDeveloper with a connection to existing DB2 database back-end to communicate with IBM Lotus Domino App via Java Web

Service to exchange static and database records. You can now run the New ADF Fusion Application, make new connection to DB2 database, Start/Run Integrated WebLogic Server, Create Java Class, Create Web Service, Test and Inspect Web Service in Browser, Deploy Web Service to Integrated WebLogic Server, Stand-By for IBM Notes Domino Designer App, then run them Side for Handshake, Communication...

Run Xpages App from the IDE, using Browser button

At this point we assume IBM Lotus Notes Domino Designer is running, a connection to IBM DB2 has been established. It is a good idea to first launch the Integrated Oracle WebLogic Server prior to Deploying applications. Follow instructions and screenshots below to create an application from scratch within minutes using the JDeveloper IDE- You should now launch the Xpage to Browser and perform a submit; areas of interest will be highlighted for your convenience.

Steps:

(1) Go to JDeveloper, Create a New ADF Fusion Application (Figures 1a, 1b)
(2) Name the application appropriately,
(3) Add proper package, example com.dokoll.solutions.inc.WebServs
(4) Choose right ADF technologies, accept defaults for Model & View Controller
(5) Create a Database connection to Allegation table in DB2 (to be used in Part 2)
(6) Create a Java Class for Static Web Service (Figures 2a, 2b)
(7) Convert Java Class into Web Service
(8) Test and Deploy Web Service (Figures 3a, 3b)
(9) Copy Web Service WSDL URL Figures 4a - 4h)
(10) Create IBM Notes Domino Consumer Service using WSDL (Figures 5a, 5d)
(11) Manage the Consumer Service Properties of Domino Designer
(12) Create JavaAgent to Consume the new Web Service (Figure 6a)
(13) Import Web Service Consumer into JavaAgent (Figure 6a)
(14) Reference WebService Call in JavaAgent code (Figure 6a)
(15) Run JavaAgent in IBM Notes Client (Figure 6d)
(16) Look in JavaAgent Console for Resultset (Figure 6d, 6e)
(17) Make Copies of Both Appplications

TechNotes: If you are here, this means you have previously gone through Part One of this tutorial- You can choose to take a look now to make more sense of the following tutorial...

Objective: Create IBM Notes Domino Consumer Web Service using WSDL. .. we will begin by first creating back-end technologies, example a form, view, JavaAgent,

then by creating a JavaBean and an Xpages file, including a configuration See **Form** and **View** design below, visit related screenshots to see how each piece connects to one another...

Step 10

Figure 5a

Excitement is mounting, now we are ready to create an application in IBM Notes Domino Designer and pull in the Web Service from Oracle Jdeveloper- We will begin by creating a form... If you need to see tutorials on how to create a form, cosult prior tutorials here or online, pretty baisc stuff, you can do it...

Figure 5b

Obviously, you also shoud create a view to help transmit data bad forth to programs... likewsie here, if you need to see tutorials on how to create a view, cosult prior tutorials here or online, pretty baisc stuff, anyone can do this, eyes shut...

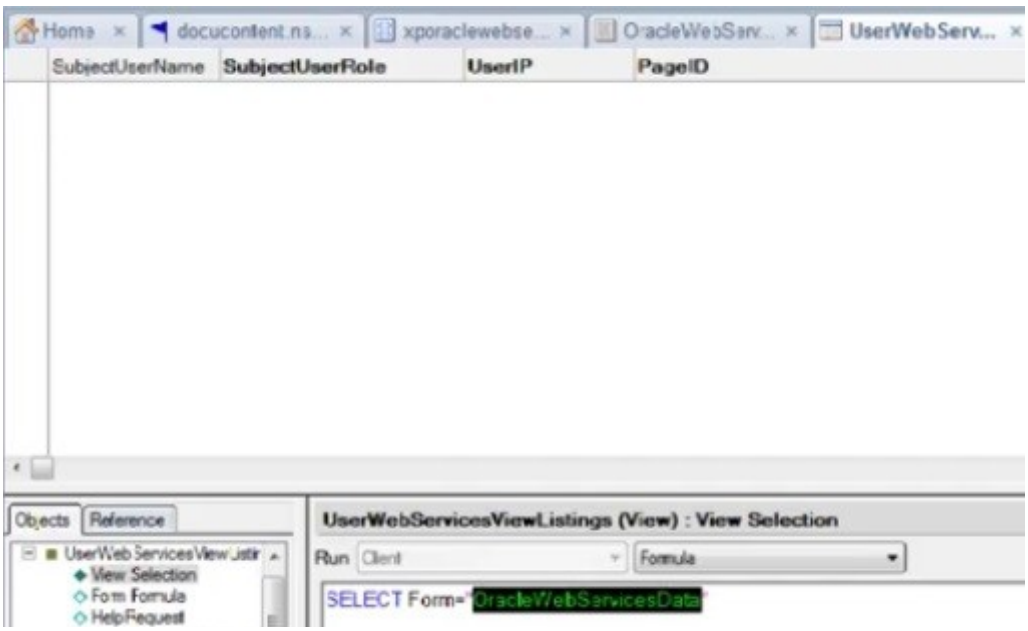

Figure 5c

You would also need an Xpage, the XML would need to be modifed to comliment Java code being used to run Web Service from Oracle Jdeveloper...

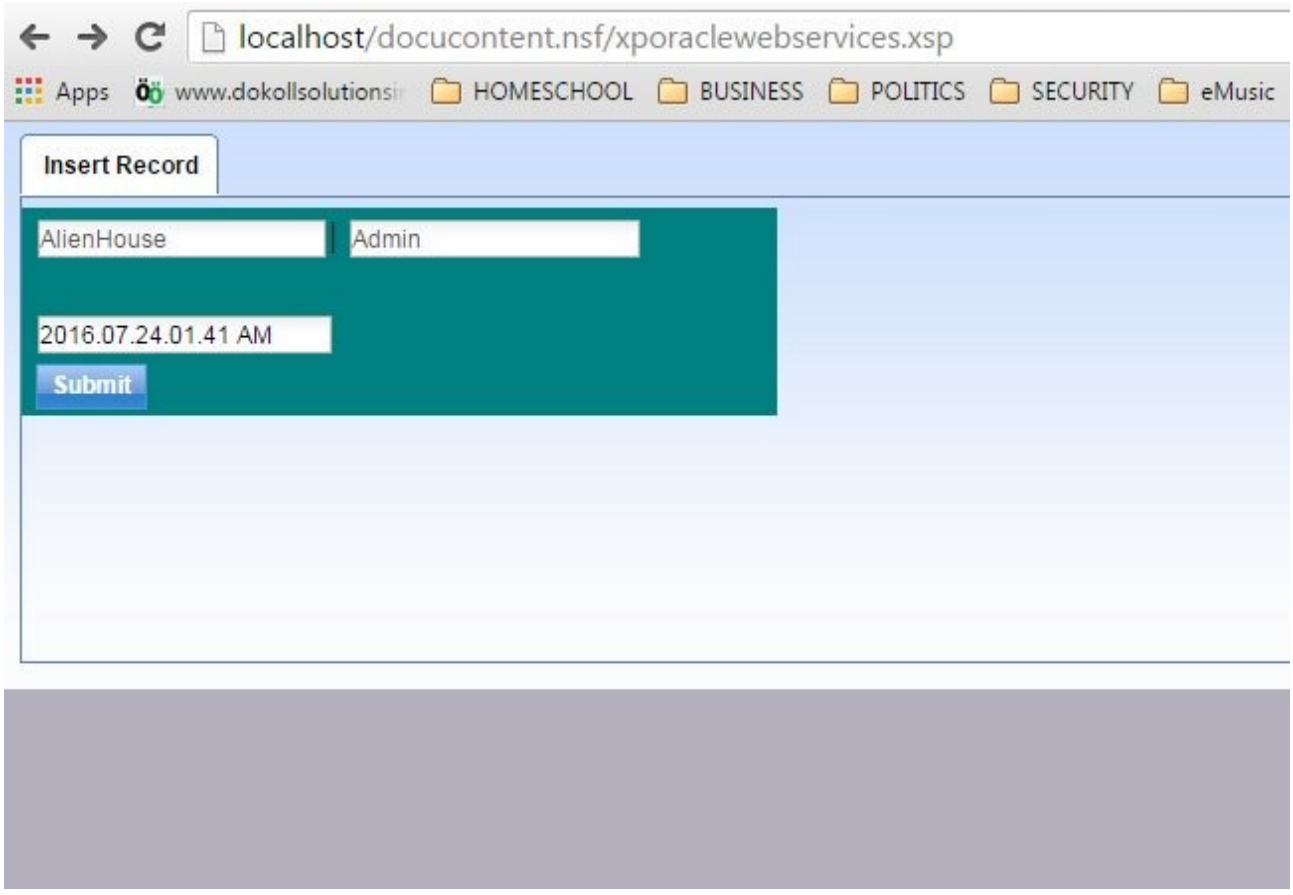

Figure 5e

Button code tied to a JavaBean with a method to run the Oralce JDeveloper Web Service, via JavaAgent in the back-end-

```
<xp:div
    style="font-size:13pt">

    <xp:button
        value="Submit" id="button1" style="font-size:9pt">
        <xp:eventHandler
            event="onclick" submit="true" refreshMode="complete"
            immediate="false" save="true" id="eventHandler1">

            <xp:this.action><![CDATA[#{javascript:SendOracleWebServiceRecordJavaBean.submitEntry()}]]></xp:this.action>
        </xp:eventHandler>
    </xp:button>
     
    <xp:messages
        layout="table"
        style="color:red;font-size:9pt;font-weight:bold"
        id="messages1" />
</xp:div>
```

Figure 5f

Configuration files (facesconfig.xml), helping to connect the JavaBean class to the Xpage, with button that runs the class...

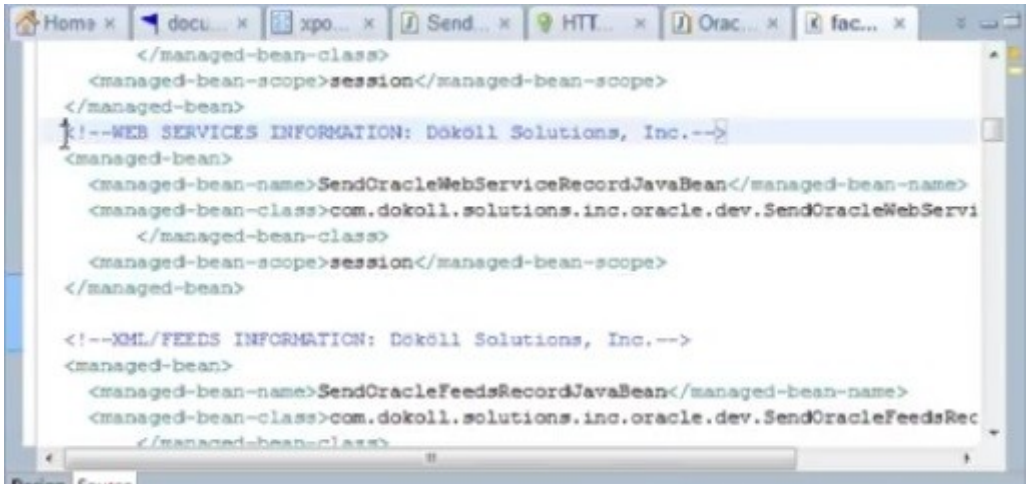

Figure 5g

Excerpt of the method that is actually doing the submission... a date value is posted to JDeveloper Web Service using Domino Xpage, JavaAgent runs the Service, which then returns the 'Date' string to Console...

```
// instantiate the document create call
Document submitDocument = (Document) database.createDocument();
// submit using GoogleSpreadsData form (Alternatively, one can add
// the Alias of that form)
submitDocument.appendItemValue("form", "OracleWebServicesData");
// using appendItemValue to insert in the fields needed
// notice we are referencing the field on the Xpage, and the
// JavaBean variables
//submitDocument.appendItemValue("subjectUserName", SubjectUserName);
//submitDocument.appendItemValue("subjectUserRole", SubjectUserRole);
//submitDocument.appendItemValue("subjectVideoID", SubjectVideoID);
//submitDocument.appendItemValue("subjectEmbedCode", SubjectEmbedCode);
//submitDocument.appendItemValue("subjectMessageBox", SubjectMessageBox);
submitDocument.appendItemValue("subjectDateCreated", SubjectDateCreated);
submitDocument.appendItemValue("userIP", UserIP);
// no need to show this on the Xpage
submitDocument.appendItemValue("pageID", PageID);

// saves the data, based on above fields to GoogleSpreadsData form
submitDocument.save();

System.out
        .println("inserting record for Oracle Web Service...");
```

Conclusion

You are now able to communicate between two different IDE via Web Services, with Oracle JDeveloper as the Provider Web Service, and IBM Notes Domino as the Recipient (Consumer)... Crafted Xpages application by first creating a form, a view, the Xpage, and added a JavaBean... Looking ahead to Web Service Console results between Oracle JDeveloper and IBM Notes Domino Designer enviroments-

Tutorials:

IBM DB2 in Oracle Products

https://www.youtube.com/watch?v=7C8mU4ap47Y&index=4&list=PLL0J_OmDhsPSWnwEXFCqkHMNjuo2p-

Oracle SQL Developer

https://www.youtube.com/watch?v=S2TcjulXij4&index=1&list=PLL0J_OmDhsPSWnwEXFCqkHMNjuo2p-UAC

Oracle Jdeveloper

https://www.youtube.com/watch?v=2kBNIaKDWwg&list=PLL0J_OmDhsPSWnwEXFCqkHMNjuo2p-UAC&index=3

Related Info:

Visit prior video tutorials to create connection to IBM DB2 Express-C

IBM DB2 Database

http://www.dokollsolutionsinc.com/CutAndPasteDB2DataLoad.html

IBM DB2 DataSource

http://www.dokollsolutionsinc.com/CutAndPasteDB2DataSourceLoad.html

For all Questions and comments, please add a Quick note to our Contact form, or visit our social media networks

Contact
http://www.dokollsolutionsinc.com/apptrendscontactemail.php

Facebook
https://www.facebook.com/Dököll-Solutions-Inc-233555900032117/

Google+
https://plus.google.com/u/0/+DököllSolutions/posts

Twitter
https://twitter.com/DokollSolutions

YouTube
https://www.youtube.com/channel/UCSImDTpK0oe7QrPsYOE4nww

Dököll Solutions, Inc.
www.dokollsolutionsinc.com
version:2016.10.02.9.00.AM