Building a File Upload Interface Using APEX Web Services and jQuery

ORACLE Gold Partner

insum
Oracle Specialists
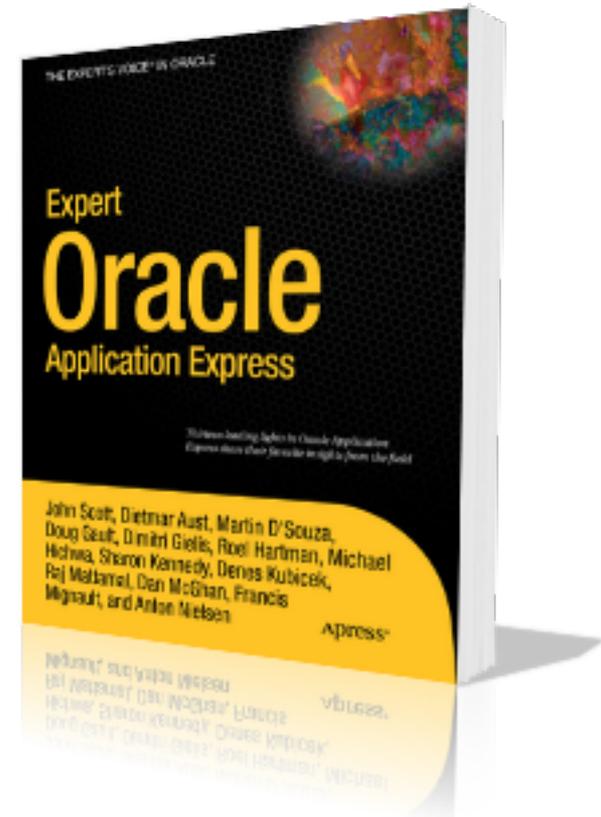## APEX Experts

insum.ca

insum

- ► Oracle APEX expert practice since 2004
- ► Oracle APEX Center of Excellence
- ► Comprehensive Consulting Services
  - ► Training & Coaching
  - ► Architecture & Analysis
  - ► Project Management
  - ► On-site / Off-site Development
  - ► APEX Best Practices and Methodologies

- ► Offices in Canada and USA
- ► Our Core Values : Respect, Teamwork, Integrity, Innovation, Quality and Fun !

ORACLE® Gold Partner

Francis Mignault

▶ CTO and Co-Founder of Insum Solutions

▶ 28 Years in IT, 25 with Oracle databases, APEX since 2004.

▶ Presenter:
  ▶ *ODTUG Kscope & APEXposed, IOUG Collaborate, Oracle OpenWorld, ORA*GE*C

▶ Co-author: Expert Oracle Application Express

▶ Twitter : @Fr4ncis

Introduction

One of our customers wanted to keep all his documents in a document management system (DMS) and wanted to use APEX to build a project management system.

Requirements :

► Integration with a Document Management System - DocuClass

► Uploading multiple files at the same time

► Drag and Drop

► Files can be big 250M +

► Fast upload

► Two senior Insum APEX Developers:

  – Credit to Reginald Horacius and Vincent Morneau

Approach

In order to be able to deliver according to the requested requirements, we tried different things :

- ► Default APEX file upload
  - – Use of the Browse button
  - – One file at a time
  - – Limitation on file size
  - – Slow
  - – Timeouts
- ► Different APEX Plugins
  - – Uploading more than 5MB , not working
  - – Changes to be made in Listener
  - – Did not work properly
  - – Not user friendly
  - – Not flexible (encrypted), cannot Customize

Approach

After about 3-4 days, we decided to use a JQuery plugin :

- ► Blueimp JQuery Plugin File Upload

  - − http://blueimp.github.io/jQuery-File-Upload/

- ► Found a blog :
  - − Monkey on Oracle - Håvard Kristiansen
  - − http://monkeyonoracle.blogspot.ca/2011/10/multiple-file-upload-with-jquery-and.html
  - − Integration of the Blueimp plugin with APEX and APEX Listener V1

Approach

► Oracle Documentation :

    – APEX Listener Documentation for RESTful APIs

    – http://docs.oracle.com/cd/E21611_01/doc.11/e21058/rest_api.htm#BGBDHHEJ

► Customized the solution with the latest versions :

    – APEX 4.2

    – Oracle 11G

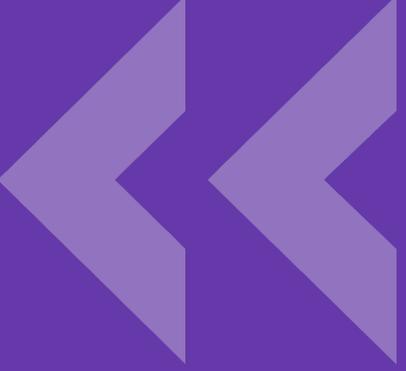   – APEX Listener V2 with Restful Webservices

Components

In order to be able to create a functional and user friendly file upload we had to integrate the following components :

- ► JQuery Plugin

- ► Restful Webservice with the APEX Listener (ORDS)

- ► HTML Form, JavaScript and CSS

- ► Modal Popup page, Skillbuilders APEX Plugin

- ► Bootstrap

And, as the final step, we uploaded the documents in the Document Management System

- ► DocuClass

Demonstration
File Upload

Walkthrough

To configure the JQuery Blueimp file upload in APEX 4:

- ▶ Create a RESTful webservice
  - – This webservice will POST / insert the image in the database
  - – Link with the APEX Listener pre-configured parameters (See doc)
    - – X-APEX-FORWARD
    - – X-APEX-STATUS-CODE

- ▶ Create an HTML Template with a Form
  - – The Form will POST to the webservice
  - – Use Skillbuilder Modal popup page

- ▶ Change the file Upload JQuery Plugin JavaScript

- ▶ Use Bootstrap for the look

Demonstration
File Upload setup

Adding parameters

To add custom parameters to the file upload:

- ► Step 1 : Create a new column in the table

- ► Step 2 : Create an APEX Item on page 2

- ► Step 3 : Change the Webservice to insert the new column and add a parameter

- ► Step 4 : Change the JQuery plugin Jquery.fileupload.js

  - – Section before send, use the parameter name

  - – xhr.setRequestHeader('x-project-name',apex.item("P2_PROJECT_NAME") .getValue());

Integration with DocuClass

Once the file had been uploaded in the database, the integration with DocuClass was fairly easy

- ► Use DocuClass APIs to upload the documents:
  - – Create a secure connection
  - – Do an HTTP Post to get a file ID
  - – Do a 'document add' with file details

```
l_response := do_post (p_parameters => 'action=document_add'
                        ||'&user_id='||g_user_id
                        ||'&request_id='||g_request_id
                        ||'&ticket_id='||g_ticket_id

              ,p_request_body => '<Request>'
                        ||'<Parameters typeId="'||p_type_id||'">'
                        ||CASE WHEN l_keywords_xml is not null THEN
                          '<Keywords>'
                          ||l_keywords_xml
                          ||'</Keywords>'
                        END
                        ||'<Files><File reference="'||p_file_ref||'"/></Files>'
                        ||'</Parameters>'
                        ||'</Request>);
```

Conclusion

Integrating APEX with different technologies was relatively easy once we figured out how the JQuery plugin and the DocuClass APIs worked.

We are looking to create an APEX plugin for that but have some limitations due to the usage of a RESTful Webservice.

In the end, the customer was very happy with the results and can now upload multiple files of different sizes in the Document Management System using APEX applications.