COLLABORATE 17
TECHNOLOGY AND APPLICATIONS FORUM FOR THE ORACLE COMMUNITY

EBS Excel Upload Options

Thu 4/6/2017, 9:45-10:45

Session ID #: 10063
Jasmine G

Prepared by: **John Peters**, JRPJR, Inc.

john.peters@jrpjr.com

www.jrpjr.com

Revision 2.0

Session ID: 10063

About the Presenter

John Peters, JRPJR, Inc.

- Independent Consultant
- Techno/Functional focus
- Supply Chain Functional focus
- Over 24 years working with Oracle EBS
- Founding board member of the NorCal OAUG
- OAUG Workflow SIG Chairman
- OAUG APEX in EBS SIG Chairman
- Member of the Oracle ATG CAB
- I enjoy sharing what I know about the Oracle E-Business Suite, I have many prior presentations posted on my web site at: http://jrpjr.com

Why Excel

Why does data end up in Excel

- External systems
  - Data Conversions
  - Interfaces
- EBS Extract manipulate upload
  - Due to a functional gap in EBS
  - Due to an inefficiency in EBS
- Excel is an easy to use and flexible tool so it is natural for users to evolve business processes around it

Drawbacks to Excel

- Lacks auditability and integrity of data
- Excel is often the bane of IT and Audit professionals
- Goals should be to:
  - Try to keep business processes in a system which provides:
    - Authentication
    - Authorization
    - Auditability
    - Structure to maintain data integrity

Excel Will Exist in Business Processes

- Okay I am getting off my Soap Box
- Excel is not going away
- So we need to find better ways to use it and integrate it into EBS

Ways to Extract from EBS to Excel

- Folder Forms, save directly to Excel
- BI Publisher, save directly to Excel
- Discoverer, TOAD
- Monarch, parse structured report data to Excel
- Third Party Reporting/Extract Tools
- Oracle APEX, save directly to Excel

Manipulate Data in Excel

- Always make sure that:
  - Data is row structured
  - You include all columns in your sorts
  - Format data with leading zeros as text
  - Get the correct date format for later parsing

- These may seem obvious but they are common issues we all run into daily when dealing with Excel data

Methods to Import Data From Excel

Methods to Import Data

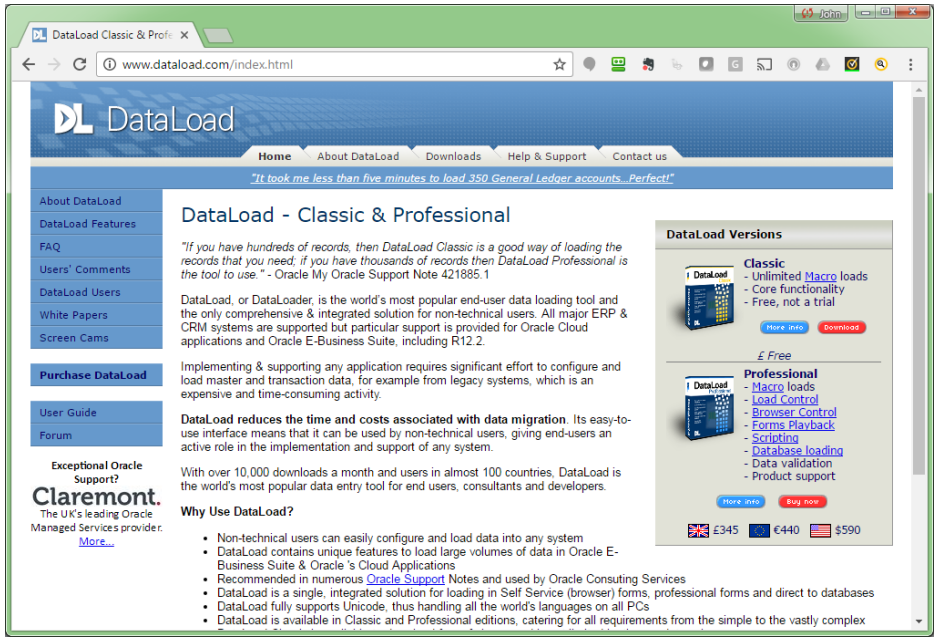
- We are going to now compare and contrast some methods for importing your Excel data back into Oracle EBS
  - DataLoader
  - SQL*Loader
  - DB External Tables
  - PL/SQL
  - Web ADI
  - Third Party Import Tools
  - APEX

DataLoader

DataLoader

- Cost: Classic (free) and Professional ($590)

- Where to get more information: http://www.dataload.com/



DataLoader – How it Works

- It is essentially formatting a stream of keystrokes that are sent into the EBS window
- Spreadsheet like interface to help structure navigation and control keystrokes along with data

DataLoader – Benefits

- Users can easily identify with how it works
- Uses same forms validations as manual entry
- Good for small number of very repetitive sets of records in a single EBS forms region
- Can load into EBS Forms and Self-Service (HTML) pages (Professional version only)

DataLoader – Drawbacks

- Not robust, just stream of keystrokes with fixed delays/pauses (Professional version has a Load Control feature)
- If system hangs, network communication delay, keystroke stream and system can get out of sync
- Keystrokes go into wrong field or just pile up in a field with an LOV that does not line up
- Multi page or region forms are tough to get to work
- Ties up a PC, can't change window focus
- Only runs in Windows OS
- Other PC application Pop-ups interfere with load because the steal the focus of the Windows UI

Oracle SQL*Loader

Oracle SQL*Loader

- Cost: Free
- Where to get more information:
  http://oracle.com
  http://support.oracle.com

- Many other web sites that provide good details

SQL*Loader – How it Works

- Requires an Oracle Home and SQL*Net
- Must have a DB User/Pwd
- Requires the data file to be loaded to be computer with the Oracle Home and SQL*Net
  - Desktop PC
  - DB or APPS Tier Server
- Reads the files and parses, either delimited or fixed space
- Transfers data directly into the database
- Reads text files, not native Excel, so you must save as a delimited file
- Control file identifies how to perform the load and mappings
- Log file records processing details
- Bad file holds records with formatting errors or that cause Oracle errors
- Discards file holds records not satisfying a WHEN clause

SQL*Loader – Benefit

- SQL*Loader has been around as long as the database
- Lots of reference information available on the internet
- Once you get it working it is reliable
- Lots of details to help with debugging
- Can handle parent child relationships in data

SQL*Loader – Drawbacks

- Many ways to parse a file and declare the parsing in the control file
- Tough to get working
- Does not handle line feeds in data fields well
- Does not always interpret quoting logic the same as Excel delimited files
- Access to servers is required or client install
- Many files to manage/review to ensure success
- Log files hard to parse in an automated manner
- Does not read native Excel files
- Many steps to process data
  - Save delimited file
  - Transfer file to computer where SQL*Loader is available
  - Run SQL*Loader
  - Review log file
- Still need to call Oracle API's or load interface tables
- Often different code set from forms validation

DB External Tables

DB External Tables

- Cost: Free
- Where to get more information:
  http://oracle.com
  http://support.oracle.com

- This is a variant of SQL*Loader

DB External Tables – How it Works

- Place a delimited or structured file on the DB Tier
- Create Table statement uses ORGANIZATION EXTERNAL clause to describe parsing and file location
- Files must be located in a filesystem that has been defined in the DB using the CREATE DIRECTORY statement
- Does not truly load the data into the DB, when accessed the data is reparsed each time
- Since it is a DB object it can be accessed by multiple sessions at the same time

DB External Tables – Benefits

- You can plug a text file into the DB and read it live like any other table
- Less steps than SQL*Loader

DB External Tables – Drawbacks

- You have to be on the DB tier, with file permissions that allow the DB to read the file
- You need DBA access to completely setup
- Very tough to debug you need DBA access
- Since it is read and parsed live, the current data in the file is what you see, with SQL*Loader you typically are loading into a staging table iteratively
- Same parsing limitations as SQL*Loader
- Does not read native Excel files
- Still need to call Oracle API's or load interface tables
- Often different code set from forms validation

PL/SQL

PL/SQL

- Cost: Free
- Where to get more information:
  http://oracle.com
  http://support.oracle.com

- You would typically do this if you don't like the parsing provided by SQL*Loader or DB External Tables

PL/SQL – How it Works

- You can use the DB Utilities in UTL_FILE
- File must be saved as delimited or fixed column text
- File must be transferred to the DB Tier
- Must first use CREATE DIRECTORY to identify where on the DB Tier the file will reside
- You then use UTL_FILE.GET_LINE to read a line
- Parsing the line into columns is your responsibility

PL/SQL – Benefits

- Complete parsing control

PL/SQL – Drawbacks

- You have to save the Excel file in a delimited format
- You have to upload the file to the DB Tier server
- Requires development

WebADI

WebADI (Web Applications Desktop Integrator)

- Cost: Free
- Where to get more information:
  http://oracle.com
  http://support.oracle.com

- http://jrpjr.com/
  Custom SubLedger Accounting JE WebADI I/F
  Custom Web ADI Integrators

- OAUG has a WebADI SIG
- Many other web sites that provide good details

WebADI – How it Works

- Data in Excel file is extracted and uploaded to the APPS Tier server, no text parsing
- Bi-directional communication between the APPS Tier server and client PC
- WebADI Excel spreadsheet is downloaded from the APPS Tier to the client PC
  - This includes the WebADI code, extracted data and connection details to reupload data
- You must set the appropriate macro security settings in Microsoft Excel for Web ADI Integrator to work with Microsoft Excel
- You can use Oracle EBS's seeded integrators or create your own custom ones
- Meta data associated with Web ADI Integrators is stored in the EBS instance

WebADI – Benefits

- Many seeded by Oracle already, you just have to download, put your data in, then upload
- You can now develop custom Web ADI integrators
- Don’t need to save delimited file and reparse
- Don’t need to have APPS or DB Tier access
- Don’t need a DB user/pwd, authentication and authorization handled within EBS
- Immediate feedback of errors
- Updates the status of the load, easy to interpret

WebADI – Drawbacks

- Authentication and instance details embedded in Excel file, you can't use them on different instances
- Two Excel file solution, you often have the data in another spreadsheet file and you need to cut/paste into Web ADI file
- Custom Web ADI integrators are a challenge to build, documentation and terminology is cryptic
- Still need to call Oracle API's or load interface tables
- Often different code set from forms validation

Third Party Tools

Third Party Tools

- More4Apps
  - End User Wizards -
  - Application Interface Wizard – for developers to create integrators
- GlobalSoftware
- Toad and SQL*Developer

- There are many more

More4Apps

- WebADI Replacement
- https://more4apps.com/
- End User Wizards
  - Prebuilt Loaders for Excel
  - Well over 30 transaction specific wizards like: Suppliers, AP Invoices, Items, BOMs, Routings, etc
  - No development required, license, install and deploy
  - Becomes an application module in EBS
- Application Interface Wizard
  - For developers to create integrators
  - Improved metadata entry and maintenance over WebADI

GlobalSoftware

- WebADI Replacement
- https://www.globalsoftwareinc.com/
- Spreadsheet Writeback for Oracle

More4Apps, GlobalSoftware - Benefits

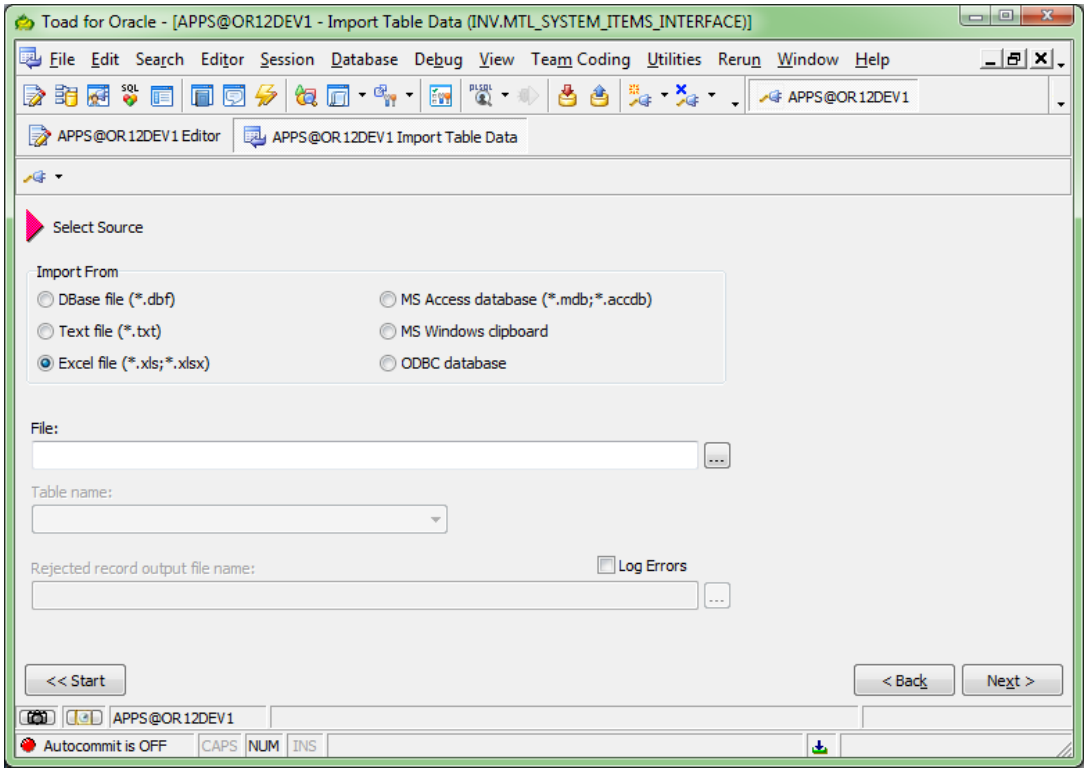
- Supported Tools
- No development when using provided integrators
- Easier to use than Custom Web ADI

More4Apps, GlobalSoftware - Drawbacks

- License expense
- Proprietary tool, limited community experience

TOAD

- TOAD has the ability to read native Excel files into a database table
- SQL*Developer has a similar functionality
- Requires Development
- Need TOAD License
- Need SQL*Net connectivity
- Need DB user / password
- Database :
  Import :
  Import Table Data



TOAD - Benefits

- Reads native Excel files
- If you are already doing development it is a quick and easy way to get data into a database table

TOAD - Drawbacks

- Only meant for developers to use
- Not an end user tool

Oracle APEX

Oracle APEX (Applications Express)

- Cost: Free
- Where to get more information:
  https://apex.oracle.com/
  http://support.oracle.com

- http://www.jrpjr.com/
  Migrate your Discover Reports to Oracle APEX
  Collaborative SIG - APEX Reporting
  Extend EBS Using Applications Express

- OAUG has a new APEX in EBS SIG
- Many other web sites that provide good details

Oracle APEX – General Information

- APEX is a web based rapid development tool
- Applications can be for Reporting and Data Entry
- Write once and deploy on any platform: Destkop, Tablet, Phone
- Always requires a live connection to an Oracle DB
- You can integrate an APEX application page into EBS using EBS form functions
- Can access EBS Authentication, Authorization and Navigation

Oracle APEX – How it Works

- Client device runs the APEX application in a browser
- APEX uses the ORDS (Oracle Rest Data Services) server
- ORDS provides an Excel parsing feature
- It can read a native Excel file with many worksheets and load the data into an Oracle Collection
- Uses the ORDS function XLS2COLLECTION, this parses the XLSX file into columns and populates the collection APEX_COLLECTIONS

Oracle APEX – Additional Details

- Oracle ORDS defaults.xml file entries
  - apex.excel2collection
  - apex.excel2collection.name
  - apex.excel2collection.onecollection
  - Apex.excel2collection.useSheetName
- Put a button in your APEX application with a Button Request Value of XLS2COLLECTION
- In your PL/SQL you loop through the table APEX_COLLECTIONS
- Read from APEX_COLLECTIONS where each Excel column is held in column c002 to c050
- Column c001 holds the worksheet name

Oracle APEX - DEMO

DEMO

Oracle APEX – Benefits

- Super simple way to upload a native Excel file from the users computer to the database
- Ability to handle multi-worksheet files
- First row contains the column names so your mapping can be automated or dynamic

Oracle APEX – Drawbacks

- You have to have APEX up and running
- You need to be familiar with development in APEX
- This is a development based solution
- Collection is limited to 50 columns, 1 is used for the worksheet name, so 49 columns of data
- I have not found a way to get the filename that was uploaded

One of Many Ways in APEX

- On Mon 4/3, Sylvain Martin with Insum presented the paper: 'When Oracle Web Applications Desktop Integrator meet Application Express'
- In that example he used the:
  - Dropzone APEX Plugin by Daniel Hochleitner
    This loads the raw XLSX file into a BLOB
  - as_read_xlsx PL/SQL Library by Anton Scheffer
    This parses the BLOB into a row of data for every cell
- This is more flexible since you are not limited by the number of columns
- This also allows you access values and formula columns
- You preserved the raw XLSX file in a BLOB column so you can always extract it again if someone wants to see the source file

Summary and Review

- We have reviewed a variety of ways to upload Excel data into the database and ultimately your EBS Applications
- Each method has some pro's and con's
- Some require development
- Some require software licenses
- The goal of this presentation was to make you aware of these various methods so you can evaluate them for your future use

Please Complete Your Session Evaluation

Evaluate this session in your COLLABORATE app.
Pull up this session and tap **"Session Evaluation"** to complete the survey.

Session ID: 10063

COLLABORATE17

TECHNOLOGY AND APPLICATIONS FORUM
FOR THE ORACLE COMMUNITY

COLLABORATE17
TECHNOLOGY AND APPLICATIONS FORUM
FOR THE ORACLE COMMUNITY

Q&A

john.peters@jrpjr.com

www.jrpjr.com