

#10383 - Fusion Applications - E-Business Suite Co-Existence: Strategy & Approach

Austin Davis and Guna Balasubramanian

Objective

The objective of this whitepaper is to demonstrate Oracle Fusion Applications as a choice for your enterprise application upgrade strategy, what makes it a good choice and the ways you can adopt Fusion Applications alongside existing legacy application and how Fusion applications co-exist with the existing Enterprise Applications.

Fusion Applications

Fusion applications, arguably the biggest Enterprise Software development of this century so far, made its first general availability in the 2011 Oracle Open world event after around five years of development effort. The suite covers a broad range of functionalities in Human capital Management, Sales, Financials, Procurement, Governance, Project portfolio management, Supply chain management with a lot more verticals to follow suit.

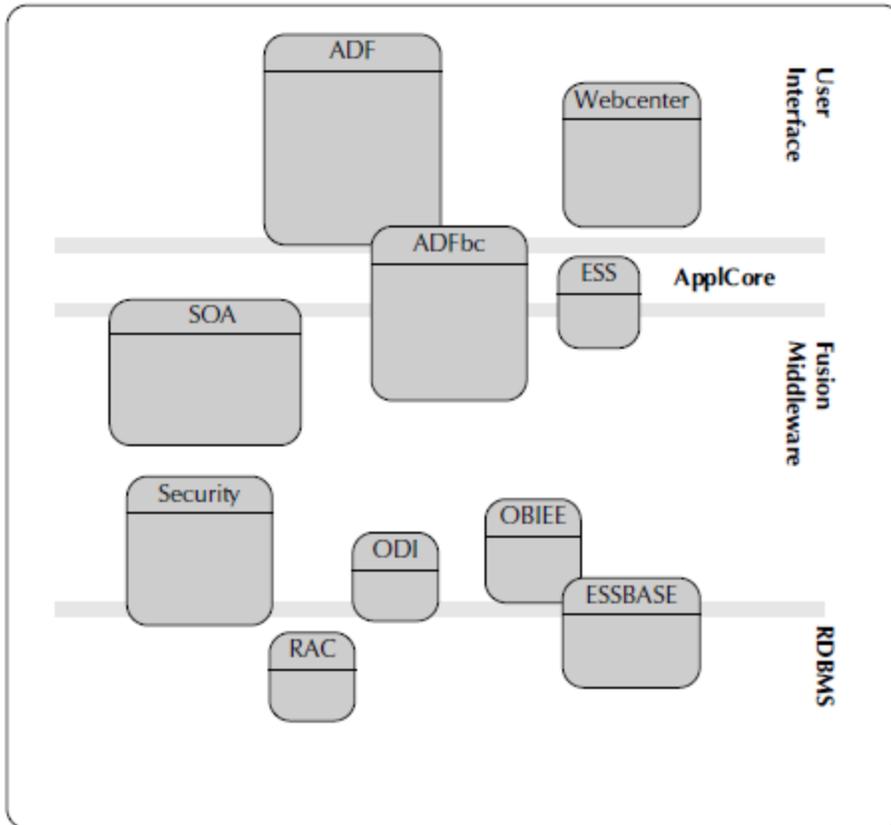
Enterprise applications as we know have a few shortcomings. They are often big, complex, with pages that maintain transactions, generates reports and offer a very limited scope for collaboration, analytics and accessibility on mobile platforms. They require a lot of investment on hardware and software and integrating these Enterprise applications requires a lot of effort with each application vendor having their own standards for integration. Enter Fusion applications, Oracle's answer to address all these shortcomings. These applications are designed and developed from the ground up to address these specific shortcomings and there are the compelling reasons why you should consider Fusion Applications in your Enterprise Application upgrade strategy.

- ≡ Enterprise Security
- ≡ Standards based Integration
- ≡ A unified User Experience across all the platforms
- ≡ Collaboration
- ≡ Ease of extensibility
- ≡ Easy deployment on cloud and on premises

These objectives each deserve a whitepaper on their own and this one would dwell deeper from an Integration Perspective, best practices and strategies for Fusion HCM and Fusion CRM.

The technology and infrastructure

Before we go further into Integration details, it would be good to have a look at the technology stack that powers the Fusion Applications as below.



- ≡ User Interface – ADF Faces & Webcenter portlets powers the User Interface of Fusion Applications. Mobile Application Framework powers the mobile Front end of the applications
- ≡ Business Services – ADF Business Components, SOA suite and Enterprise Scheduler Service makes up the Business Services layer
- ≡ Fusion Middleware – OAM, OBIEE, weblogic and Enterprise manager provides the necessary middleware infrastructure for hosting the application
- ≡ Data Store – Oracle RDBMS is the datastore of choice and webcenter UCM is leveraged for storing unstructured data like documents, wikis, web pages, attachments

Integration assets

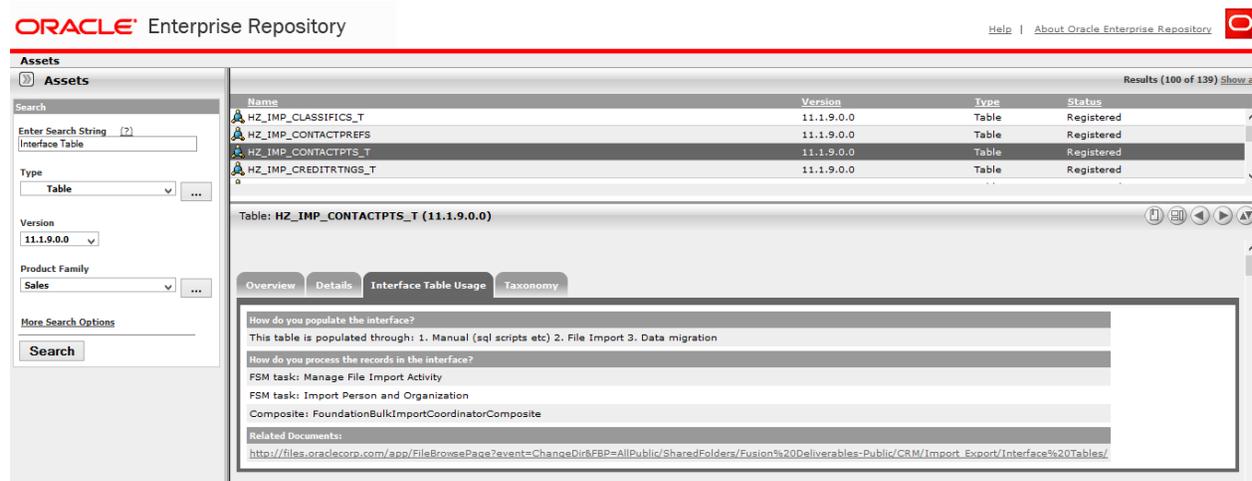
Oracle Enterprise Repository for Fusion Applications is one central place where you can find all the assets like webservices, composites etc., exposed for integration activities. There is an Oracle hosted version of the repository available in <https://fusionappsoer.oracle.com/oer/index.jsp> where we can find the standard, non-implementation specific Fusion Objects for Integration. The customer deployed OER instance provides integration assets with deployment specific metadata like flexfields, additional attributes etc., included.

Types of Integration assets

The important types of assets for Integration are given below. There are also other assets like lookups,profile options,reports that can be found in the enterprise repository **Web Services** – Fusion Applications use the SOAP/XML based webservices for exposing inbound APIs. There are two types of web services here, services exposed by ADF Business Components,Services exposed by SOA composites. ADF Services expose the CRUD functionality along with object specific operations of the underlying Business components. They also act as wrapper services for batch jobs. Composite services expose the end-points of Process flows that are typically invoked by Event Delivery Network component in SOA suite. Support for RESTful web services is brought for a few selected objects in HCM and more objects are planned for RESTful support in future.

Interface Tables and scheduled processes

Interface tables are used for bulk data loading mechanisms. Various ways in which the interfaced table can be populated are through Oracle Data Integrator, Enterprise Scheduler service,File based loader in HCM,manual(sql scripts).To find the Interface tables you can use the search parameters as shown below. The ER also has links to the documentation on how to use the Interface table to populate data. The interface tables are populated as part of a batch job controlled by the Enterprise scheduler service that takes care of moving the data from interface tables to the application.



The screenshot displays the Oracle Enterprise Repository search results for 'Interface Table'. The search criteria include 'Type: Table', 'Version: 11.1.9.0.0', and 'Product Family: Sales'. The results table lists four tables: HZ_IMP_CLASSIFICS_T, HZ_IMP_CONTACTPREFS, HZ_IMP_CONTACTPTS_T, and HZ_IMP_CREDITRTNGS_T, all with a status of 'Registered'. Below the table, the 'Interface Table Usage' tab is selected, showing documentation on how to populate the interface table and process records, along with related documents.

Name	Version	Type	Status
HZ_IMP_CLASSIFICS_T	11.1.9.0.0	Table	Registered
HZ_IMP_CONTACTPREFS	11.1.9.0.0	Table	Registered
HZ_IMP_CONTACTPTS_T	11.1.9.0.0	Table	Registered
HZ_IMP_CREDITRTNGS_T	11.1.9.0.0	Table	Registered

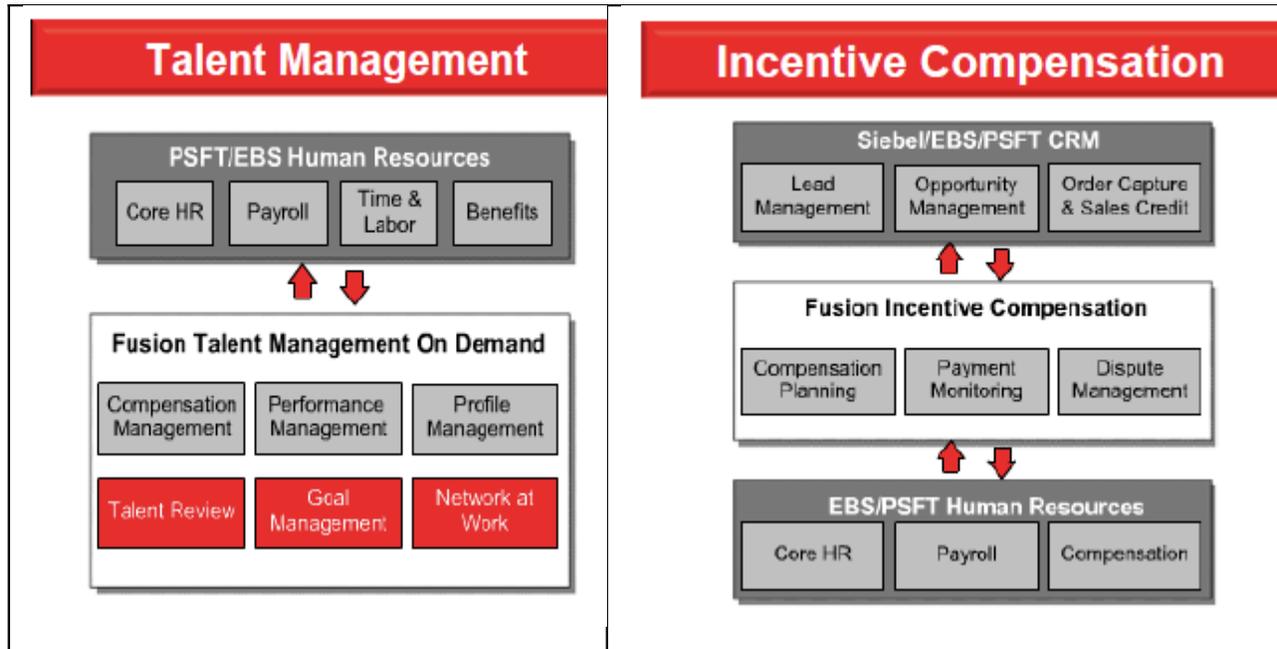
File Based Data Import

This asset provides us with object specific excel templates that can be used to populate with the data to be imported and generate a CSV using a macro. An import activity is created for the specific object with the generated CSV file. When this import activity is scheduled and run it populates the interface tables of the specific object and upon successful completion,base tables are populated.

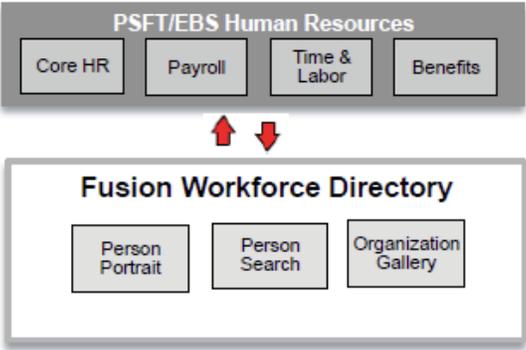
Fusion HCM Co-existence

A fast and low-risk option to adopt Fusion HCM would be the co-existence option with EBS,Peoplesoft or other third party solutions. The most commonly adopted

functionalities in Fusion are Talent management, WorkForce analytics, benefits and compensation with EBS/Peoplesoft as the System of record.



Workforce Directory



+

Fusion HCM inbound Integration

The options for inbound integration of Fusion HCM are as follows

File Based Loader – A generic load utility to bulk load data from any source into Oracle Fusion HCM objects along with object history data. Supports a limited number of objects in HCM. Supports loading data into Flexfields.

Spreadsheet Loader – Supports loading data from excel sheets. support for limited objects in HCM. It does not support all the objects supported by FBL and loading into Descriptive Flexfields is not possible through the spreadsheet loader.

Payroll Batch Loader – Specialized spreadsheet loader to load data into payroll, compensation objects.

[A closer look at File Based Loader](#)

File based loader(FBL) is used to bulk load data from any source into Oracle Fusion HCM objects along with object history data. The File based loader is used by Oracle Fusion Coexistence for HCM, the Oracle Fusion Enterprise Structures Configurator (ESC), and the Oracle Fusion HCM Spreadsheet Data Loader to load data into Fusion HCM.FBL works by preparing the data to be loaded in staging tables in object specific batches. FBL has a couple of limitations that the records can not be deleted using File based Loader and attachments can not be uploaded using File Based Loader

[FBL - Supported Objects](#)

Actions

Action Reasons

Location

Business Unit

Grade

Grade Rate

Job Family

Job

Salary Basis

Establishment

Rating Model

Talent Profile Content Item

Talent Profile Content Item

Relationship

Person

Person Contacts

Person Documentation

Department

Position

Work Relationship

Salary

Element Entry

[Support for Flexfields](#)

Loading data into flexfields is supported for the objects below.

- ≡ Assignment
- ≡ Department
- ≡ Grade
- ≡ Job
- ≡ Location
- ≡ Person
- ≡ Person Ethnicity
- ≡ Person Citizenship

Usage Scenarios

FBL can be used in full implementation to migrate data in various stages of implementation.

It also can be used in a co-existence scenario where data is shared between the source system and Fusion Application. In a coexistence scenario the following activities must be done.

- ≡ Mapping must be defined between source data and Fusion HCM
- ≡ Data extract in the source system must be managed
- ≡ Complete business objects need not be uploaded except for the Work relationships object.
- ≡ A reference implementation for co-existence scenario is given in oracle knowledge base document [1556687.1](#)

Security

The duty role File Import and Export Management Duty is required to load data via webcenter content to staging tables

This duty role is inherited by the two job roles below.

- ≡ Human Capital Management Application Administrator
- ≡ Human Capital Management Integration Specialist

To move data from staging tables to Fusion application tables, the duty role HCM Batch Data Loading Duty role needs to be assigned. This duty role is inherited by the job roles as below.

- ≡ Application Implementation Consultant
- ≡ Human Capital Management Application Administrator

Implementation best practices

Preparing the Source data

The business objects that need to be updated via the upload needs to be identified. The source data for these objects need to be verified for accuracy and upto date. There should be provision for data correction if the data is not correct. Cleaning up the legacy data would minimize the problems that could occur when loading into Fusion HCM

- ≡ Ensure that a manager exists for every worker and the information is accurate.
- ≡ For jobs and positions, the existence of accurate job codes and titles in the legacy system needs to be ensured.
- ≡ For job history, the accuracy of data should be established and the decision to move all the data or key events such as hire, promotion, termination etc., needs to be decided.

Preparation for upload

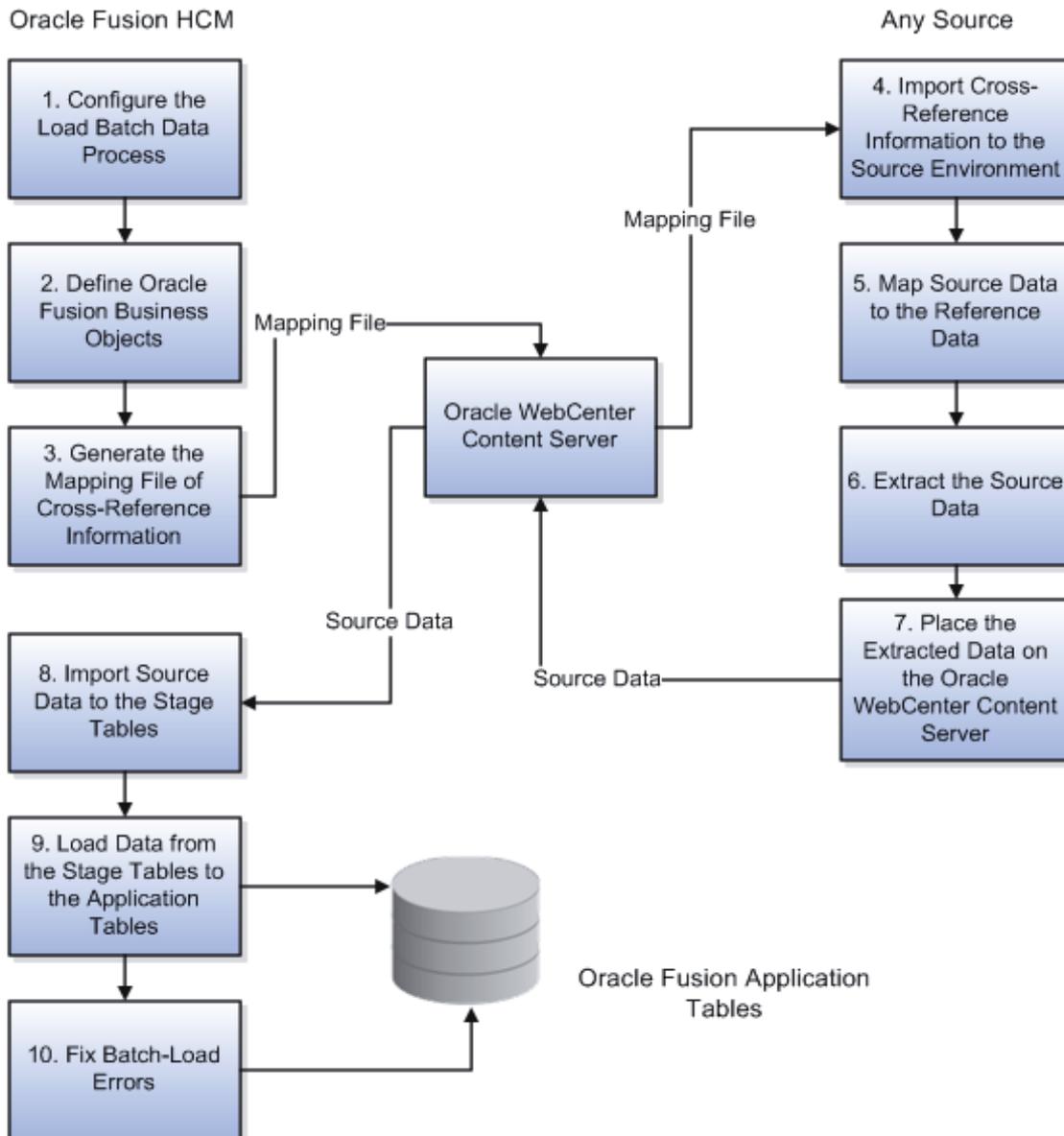
- ≡ The configuration parameters for Load Batch Data process needs to be setup
- ≡ The legal employers, business units, reference data sets needs to be identified.
- ≡ Identify the required functional mappings and ensure that they are set.
- ≡ The data transformations into FBL compatible data file format defined for various objects should be designed and used for generating the data file from source system

Managing the upload process

- ≡ The data loading should always be tested in a production clone before loading into production instance
- ≡ There are a certain validations that the legacy data needs to pass to successfully insert records into staging table. The objects and their respective validation rules can be found in the [File Based Loader for HCM Release 7 Data Loader Columns Spreadsheet](#). A validator utility based on the above spreadsheet is available as given in [1587716.1](#)
- ≡ The objects should be loaded in the prescribed order to avoid data dependency.
- ≡ Each business unit must be mapped to a reference data set immediately after loading.
- ≡ HCM File based loader and HCM Spreadsheet should not be used together to load data in a single instance.

Importing and Loading data to Fusion HCM

The process involved in loading data through File based loader is as given below.



Import Source Data to the Stage Tables

Once the zip file containing the dat files is placed in the **hcm/dataloader/import** account, we can either import or import and load the data. This can be done by either of two ways below.

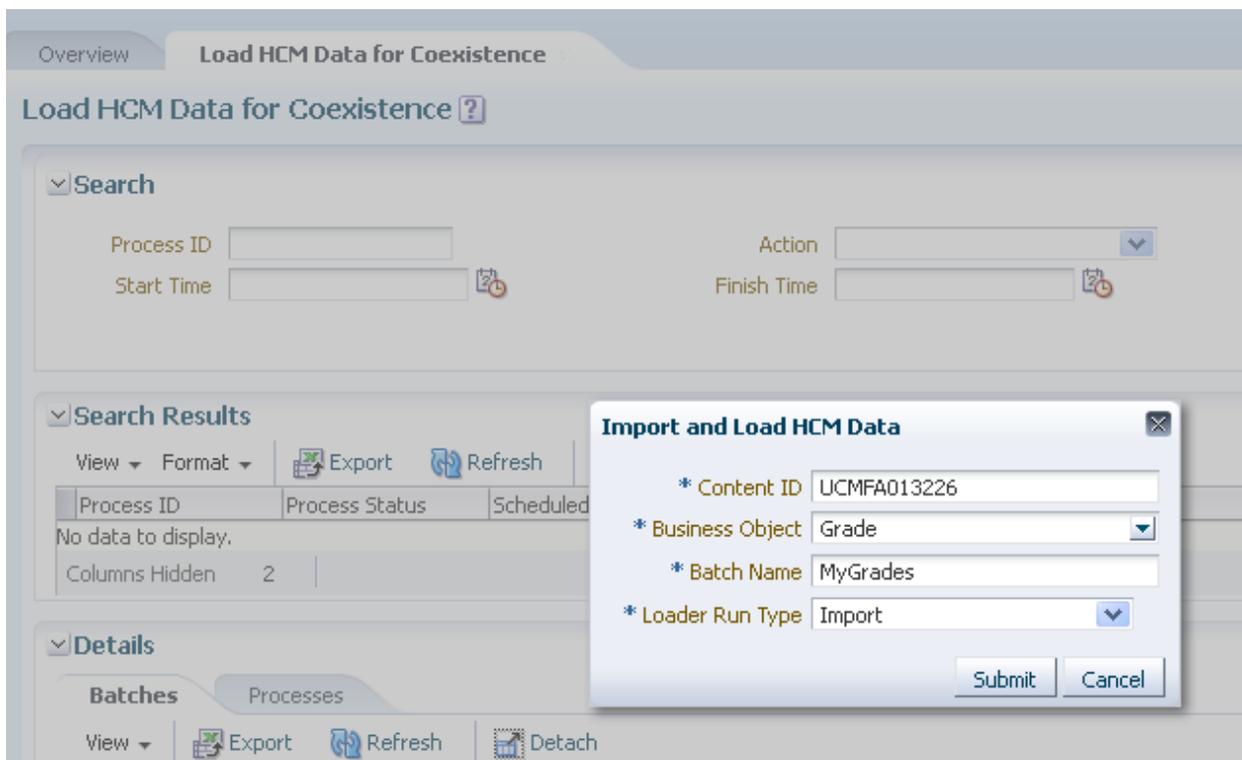
- ≡ From the Load HCM Data for Coexistence page
- ≡ Using the LoaderIntegrationService web service

Importing and Loading from the Load HCM Data for Coexistence Page

This function is delivered in Oracle Fusion HCM Release 7 Patch Bundle 3. To import or import and load a zip file from the **hcm/dataloader/import** account on the WebCenter Content server:

1. Open the Data Exchange work area (**Navigator - Workforce Management - Data Exchange**).
2. In the Data Exchange work area, select the task Load HCM Data for Coexistence.
3. On the Load HCM Data for Coexistence page, click **Import**. The **Import and Load HCM Data** dialog box opens.
4. In the **Import and Load HCM Data** dialog box, enter the content ID that you obtained when loading the file to the WebCenter Content server using the File Import and Export interface.
5. Select an individual business object or **All** to load all business objects from the zip file.
6. Provide a meaningful batch name. Object names are prefixed with the batch name to provide a unique batch name for each batch.
7. If you set the **Loader Run Type** parameter to **Import**, then data is imported to the stage tables. You can review the results of this process and correct any import errors before proceeding with the load to the application tables. When you first start to use File-Based Loader, this is the recommended approach.

If you set the **Loader Run Type** parameter to **Import and Load Batch Data**, then data is imported to the stage tables. All objects imported successfully to the stage tables are then loaded automatically to the application tables. You may prefer this approach when import errors are few and your data-loading is routine.



8. Click **Submit**.

Importing and Loading Using the Loader Integration Service Web Service

Loader Integration Service is the webservice that is used to programmatically invoke the File Based Loader operations. The parameters for the submitBatch method that is used to load the data into Fusion HCM is as shown below.

Parameter	Description
ZipFileName	Content ID of the file on the WebCenter Content server (the same value as dDocName in the WebCenter Content Java call)
BusinessObjectList	Name of the business object to be loaded. Repeat this tag for each business object to be loaded.
BatchName	Name of the batch when it is created in Oracle Fusion.
LoadType	The type of load. Can be either FBL or HR2HR. Use FBL.
AutoLoad	Indicate whether to load the data into Oracle Fusion. N = Import only Y = Import and Load This parameter in the service replaces the setup parameter Load HCM Data Files Automatically on the Manage HCM Configuration for Coexistence page.

Implications of Security Policy on the LoaderIntegrationService

The LoaderIntegrationService is secured using the following policy:

oracle/wss11_saml_or_username_token_with_message_protection_service_policy

Therefore, when a client calls the service it must satisfy the message-protection policy to ensure that the payload is transported encrypted or sent over the SSL transport layer.

Batch Load Summary Report - Monitoring and correcting errors

This extract details the business object instance that was in error. It reports the object GUID, its Fusion ID, and the detailed error that occurred for each row during the batch load process. Batch Load Summary is delivered as a seeded extract which reports on the HRC_LOADER_BATCH_LINES table with the line status as "Error in Row".

PARAMETER	DESCRIPTION
Zip File Name	Name of the file from which the batch is loaded. It is populated in the

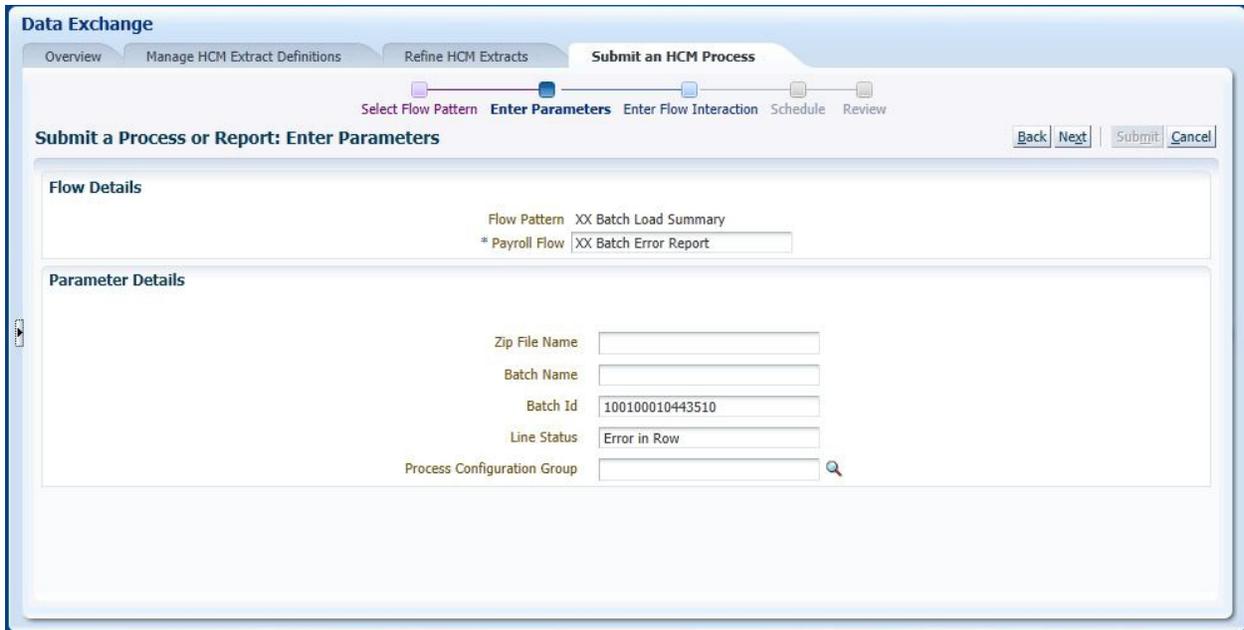
	PARAM_VALUE column of HRC_XFER_JOB_PARAMS table. You can find the value by looking at the Content ID column in File Import and Export screen
Batch Name	Name of the batch that gets generated on submission of a batch to load. It is populated in the BATCH_NAME column of HRC_LOADER_BATCH_HEADER table.
Batch Id	The Batch Loader Id that gets assigned to the Batch.
Line Status	<p>The status of the Batch as per the HRC_LOADER_BATCH_LINES table. This parameter can take any of the following values</p> <ul style="list-style-type: none"> ≡ Error in Row ≡ Ignore ≡ New ≡ Pending Action ≡ Ready to Process ≡ Successful <p>This parameter is currently defaulted to “Error in Row” in order to pick up only batch lines in error</p>

Run the Extract

1. Navigate to the Data Exchange work area.
2. In Tasks Pane, select Submit an HCM Process and Search /Select the extract defined earlier.

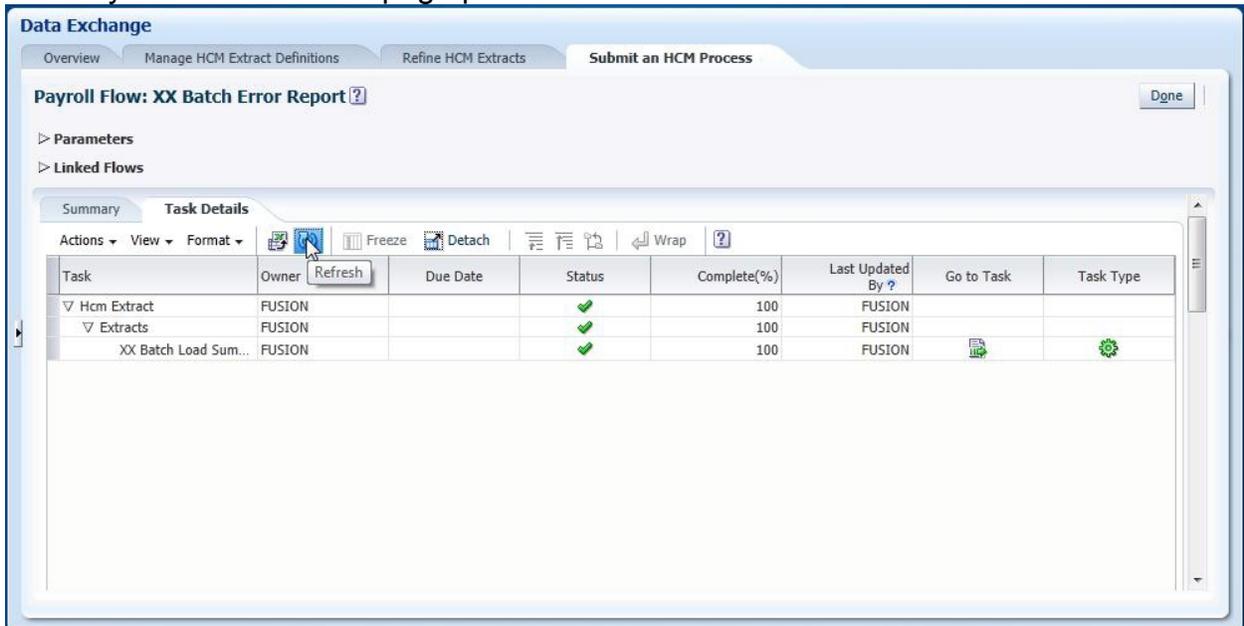
The Enter Parameters page appears.

- a. Provide a name in the Payroll Flow field for the extract you want to create [Note: This name is referred to when running the subsequent report.]
- b. Enter a value for one or more of the following parameters [details of each provided earlier in this document].
 - ≡ Zip File Name
 - ≡ Batch Name
 - ≡ Batch Id
- c. Leave the default value Error in Row in the Line Status field.



3. Click Next until you see the Schedule page.
4. Provide the desired schedule parameter, review, and submit .

The Payroll Flow checklist page provides status information on the submitted extract.



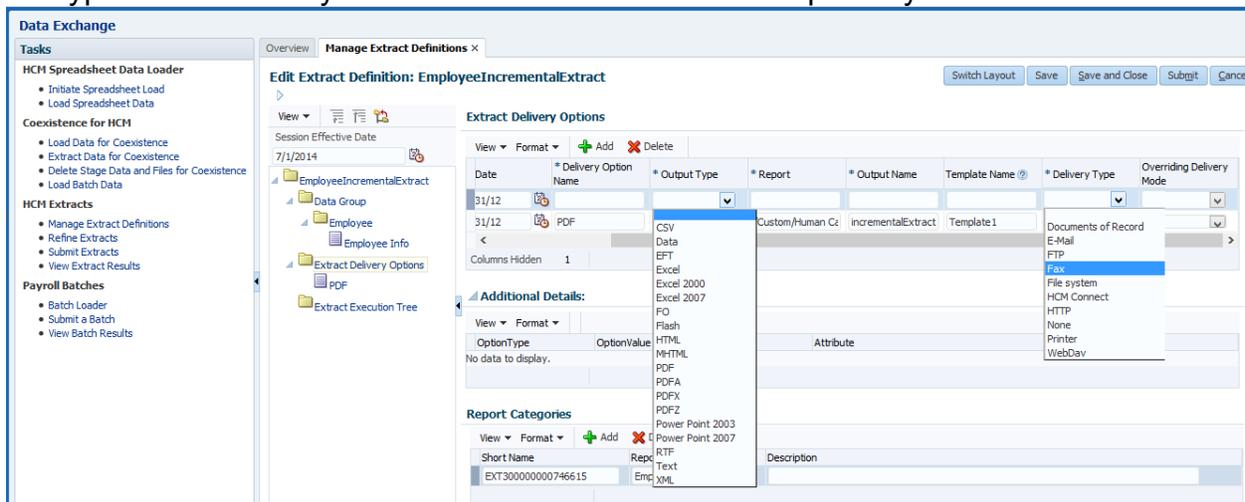
If the status shows an error, navigate to Schedule Processes in the Tools section and review the log. You can use the Go to Task link to view the extract if a BIP Template and Delivery Options has optionally been configured in the System Extract definition. It is also possible to automate the extract process through a web service. For more information pls refer to the Outbound Integration section below.

Fusion HCM Outbound Integrations

The options for outbound integrations in HCM are as follows

- ≡ HCM Extract - Extract, Transform and Format large volume of complex data.
- ≡ BI Publisher - Extract Data from HCM Business Object Repository in pre-formatted Templates. Simple queries and objects that are not supported by HCM Extracts can be integrated using this option
- ≡ OTBI - Extract Data from HCM Business Object repository to provide real-time operational reporting and file creation.

The HCM extract is a data extraction tools that can handle complex extract and reporting needs. It can be used to extract data from simple objects to hierarchial ones, extract changes only data and supports various output types like xml, csv data etc. Various delivery types like email, FTP, WebDAV, HTTP are also supported. Data encryption is supported in the 'HCM Connect' Delivery type in which the extract is encrypted and securely transferred to the Oracle UCM repository.



Automation of HCM Extract

When used as part of an integrated process it may be required to invoke the HCM Extract automatically as part of an automated flow. In addition once the extract is complete it is likely that the output needs to be retrieved for subsequent processing. When an HCM Extract is defined it generates a Flow object in the Payroll Flow data model with the same name as given to the HCM Extract itself. This flow is used to control the processing of the HCM extract including setting any parameters and handling the individual steps within the flow. By default there is just one step (Task) within the flow, but if required additional tasks can be added. When a specific HCM Extract flow is invoked a Flow Instance is created. The ability to manage the invocation of a Flow and the Tasks within it is the key to being able to automate a process through the Payroll Flow Engine, and hence the way HCM Extract can be automated.

A Payroll Flow can be invoked from outside of Fusion using the FlowActionsService, hence this will be the mechanism by which HCM Extracts can be automated. The method to invoke the HCM Extract is FlowActionsService.submitFlow().

This has the following parameters:

Parameter Name	Description
Flow Name	The name of the HCM Extract
Flow Instance Name	The identifier of the particular instance of the extract to be run
Legislative Data Group Name	The name of the legislative data group
Recurring Flag	Set to "false"
Parameters	Name value pairs of parameters. For default HCM Extract only Effective Date is required.

Example

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:typ="http://xmlns.oracle.com/apps/hcm/processFlows/core/flowActionsService/types/"
xmlns:flow="http://xmlns.oracle.com/apps/hcm/processFlows/core/flowControllerService/">
<soapenv:Header/>
<soapenv:Body>
<typ:submitFlow>
<typ:flowName>ExtractExample</typ:flowName>
<typ:flowInstanceName>ExtractExample_Run1</typ:flowInstanceName>
<typ:legislativeDataGroupName>US Legislative Data Group</typ:legislativeDataGroupName>
<typ:recurringFlag>false</typ:recurringFlag>
<typ:parameterValues>
<flow:ParameterName>Effective Date</flow:ParameterName>
<flow:ParameterValue>2014-01-01</flow:ParameterValue>
</typ:parameterValues>
</typ:submitFlow>
</soapenv:Body>
</soapenv:Envelope>

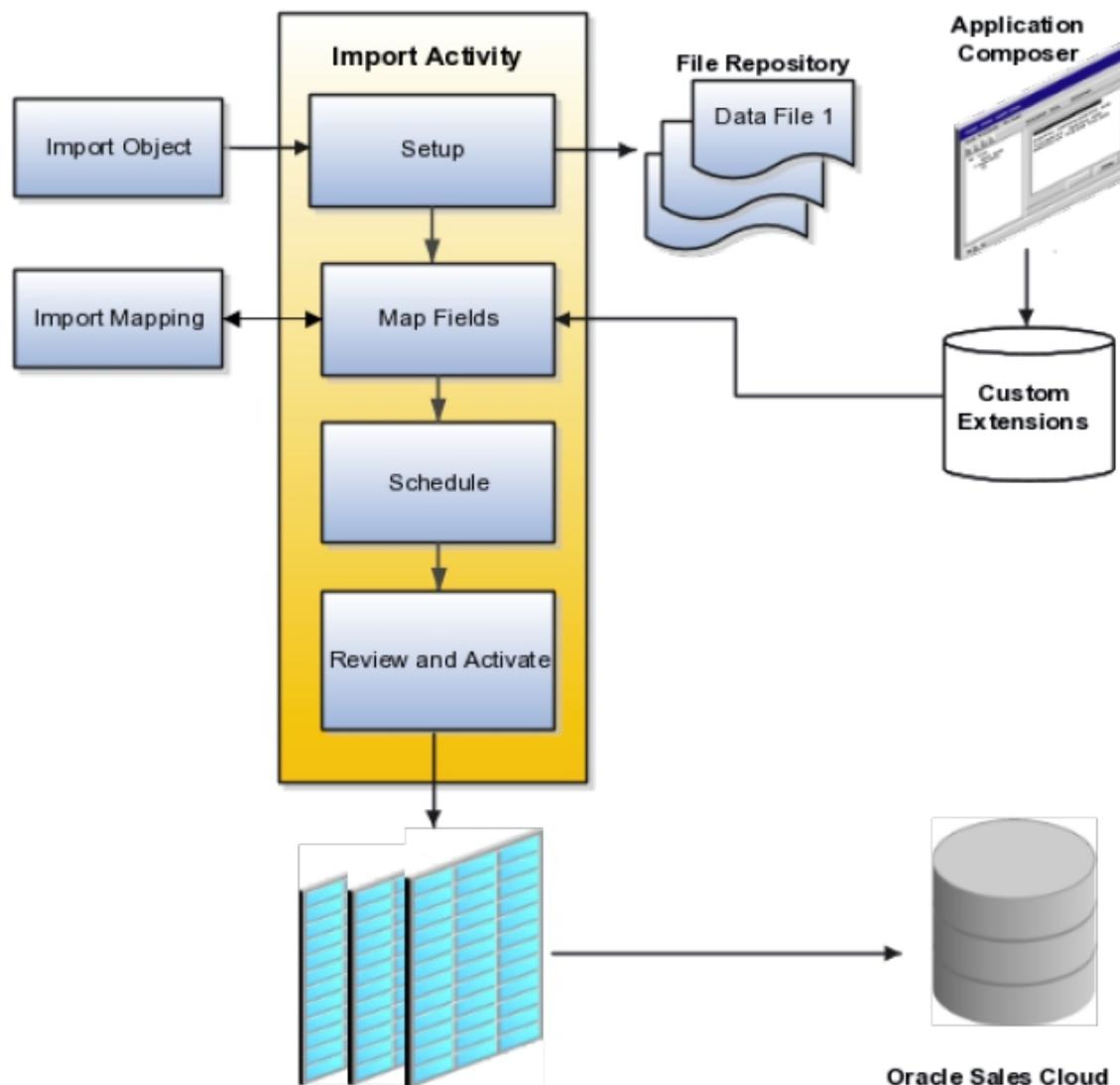
```

Fusion Sales cloud Integration

The batch integration assets for Fusion Sales cloud are File based Data import, File based Data export and Bulk export. For event oriented integrations like creating a quote from an opportunity, converting a lead into an opportunity ADF web services exposed by the individual objects can be used.

File based Data import

The diagram below depicts the process flow involved in File Based Data import.



In file based data import, an excel template would be provided for each supported object. The template is to be filled up with the data for the object to be loaded and upon successful completion of validations a CSV can be generated from this template. The user has to create a File Based Data import task with the generated csv, the object to load as the major inputs. The CSV can be either from the desktop, URL or a networked resource for automated loading. The Assets, opportunities, partners, leads, activities objects support loading attachments also. The attachment has to be uploaded in a separate zip or jar file. Once the import activity is completed, mappings need to be defined for the columns in the csv with the attributes in the object. The standard template as well as the seeded mapping are customizable. Once the mapping is complete we can create an Enterprise scheduler job to load the interface tables with the data in csv and upon successful completion of validation to the Sales cloud base tables. The File import web service exposes the service methods to manage the File import activities. The File Import Mapping Setup web service exposes methods to submit the import activity and get the status of the activity. The MetadataPublicService method can

be used to get the object definitions along with any customizations that have been made. The Enterprise scheduler service methods also can be used to manage the scheduling of import and export activities.

File Based Data export

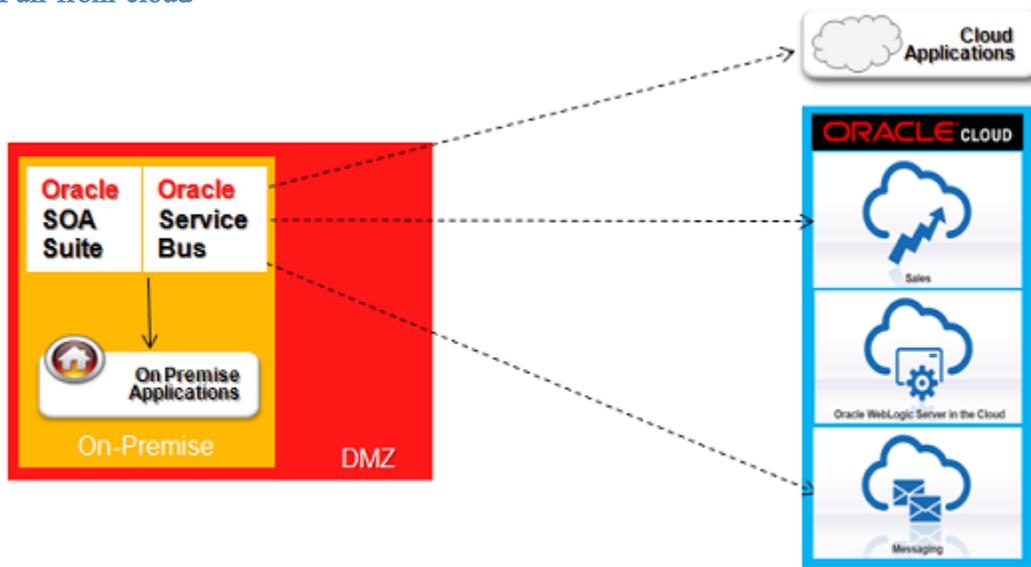
It is one of the options to export data from Oracle Sales cloud. Supports master-detail relationships though the filter criterias for exported records can be applied only at the master level. The extract output is a set of multiple csvs for master-child objects archived into a zip file.

Bulk Data Export

It is different from File based Data export in that filter criterias can be configured at child level too. It is used to extract bulk loads of data from Sales cloud. It can extract either full set records for an object or do incremental extracts. The output type of a Bulk extract is a comma separated or tab delimited files attached to the extract process.

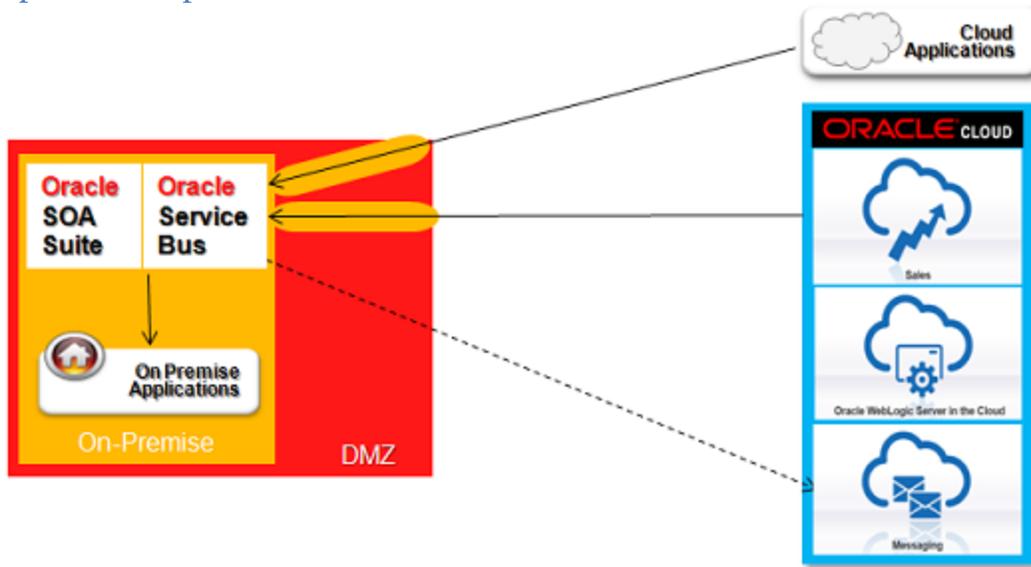
On-premise to cloud Connectivity patterns

Pull from cloud



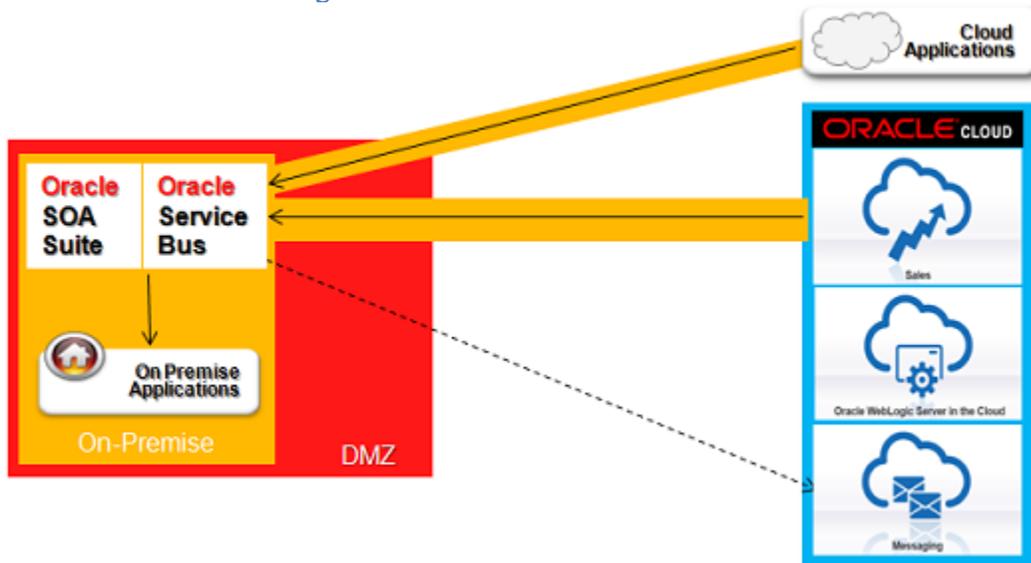
The on-premise system polls at regular intervals to pickup messages from the SAAS application. This is can be done by SOAP APIs. This way, only outbound connections happen on premise assets are not exposed outside. But polling is not suitable for all scenarios especially in batch mode and performance is affected.

Open Firewall ports



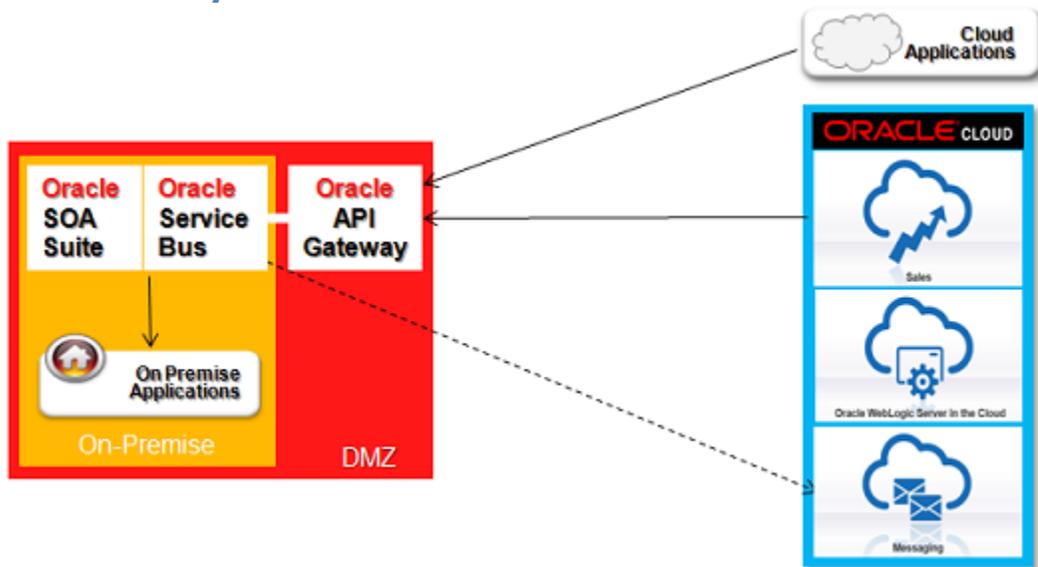
The on-premise system exposes webservices through ports opened in the firewall. This approach is used by Fusion Applications. It is much suitable for real time integration needs though Batch data processing would still be a performance bottleneck.

Virtual Private Networking



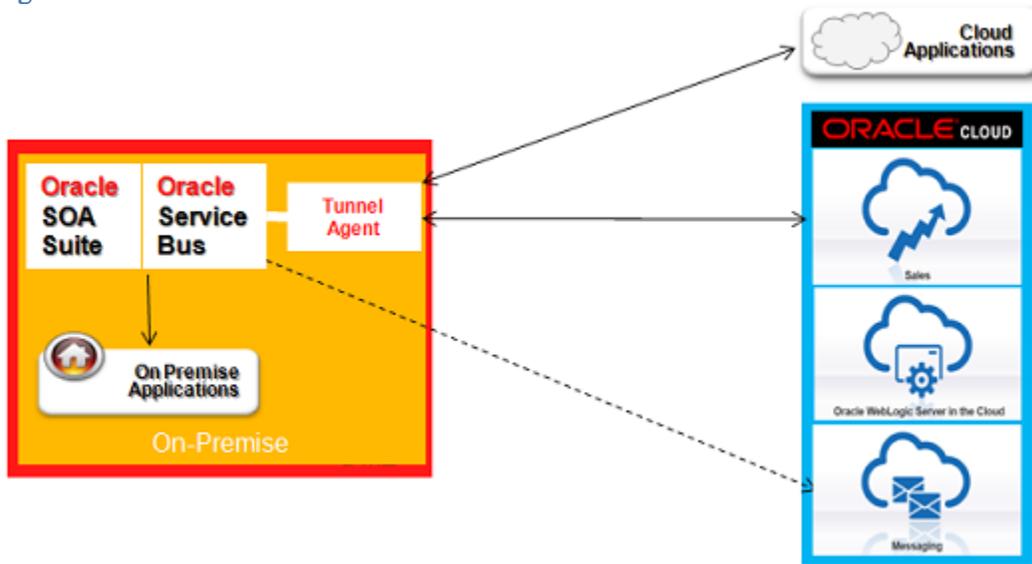
A trusted channel is established between On-premise and cloud apps by extending the network to the cloud through Virtual private networking. Bulk data integration becomes easier and security risks are minimized in this approach though it requires complex VPN to be configured, the cloud provider to support it etc.,

Reverse Proxy



A reverse proxy like Oracle API Gateway is used by the on-premise system to receive the messages. Oracle Service Bus, Oracle HTTP server virtual hosts also support reverse proxying, but without a firewall. This approach may require new software to be bought at additional cost.

Tunnelling



A light weight tunneling software mediates the communication with the cloud and the on-premise system by using the http protocol. Websockets, HTTP SSH tunneling are some known implementations. This approach has some security risks though as it can bypass firewalls, virus scans etc.,

Conclusion

The future of successfully using Fusion and Oracle eBS application product and technology lies in the ability for an organization to assess and plan both from a solution as well as a functionality perspective. For each customer the correct strategy in the use of Fusion applications can be a different journey depending on the current architectural footprint, current and future infrastructure requirements and rich business functionality. In each unique case, different risks and mitigation strategies will evolve with varying cost factors as organizations strive to stretch their bandwidth and ROI dollars. Most will benefit from a coexistence strategy while some who remain in legacy releases may benefit from a reimplementation approach—especially those moving to an Infrastructure as a Service (IaaS) and Oracle Cloud deployment model. Leveraging the existing solution with Fusion in an initial coexistence strategy would ensure no loss of functionality while gaining the benefits of reduced risk and costs associated with application and technology maintenance. This would lead to a phased deployment approach for migration of a solution from a legacy application to the Fusion platform based on availability for your application requirements. With any application and technology change engaging the organization's key stakeholders and business leaders early in the journey will prove invaluable to ensuring that the end result is a rewarding one.

Bibliography

1. Oracle Fusion Human Capital Management File-Based Loader - [1595283.1](#)
2. Oracle Enterprise Repository (OER) for Fusion Applications - <http://fusionappsoer.oracle.com>
3. [Integrating with Oracle HCM](#)
4. Integrating Oracle Sales cloud
5. [Oracle Sales Cloud File-Based Data Import Guide](#)
6. Cloud to on-premise connectivity patterns ([Rajesh Raheja](#))