**Conversion Design and Development Document**

**Human Capital Management - Employee Conversion from UCPath**

<Date>

Document Control Information

Document Information

|  |  |
| --- | --- |
| Document ID | **AE\_CNV\_RTR\_003** |
| Document Name |  |
| Document Author |  |
| Document Version | **V.1** |
| Document Status |  |

Document Edit History

| Version | Date | Additions/Modifications | Prepared/Revised by |
| --- | --- | --- | --- |
| 1.0 | 25-07-2022 | Added additional step of linking user account and role to all employees | Dheera Paul |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Document Review/Approval History

| Date | Name | Organization/Title | Comments |
| --- | --- | --- | --- |
| <DD-MM-YYYY> |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Distribution of Final Document

The following people are designated recipients of the final version of this document:

| Name | Organization/Title |
| --- | --- |
|  |  |
|  |  |
|  |  |

Table of Contents

[1 Introduction 4](#_Toc85529167)

[1.1 Objective 4](#_Toc85529168)

[1.2 Dependencies and Prerequisites 4](#_Toc85529169)

[2 Conversion Description 5](#_Toc85529170)

[2.1 Conversion Scope 5](#_Toc85529171)

[2.2 Assumptions 5](#_Toc85529172)

[2.3 Proposed Process & Ownership 5](#_Toc85529173)

[3 Conversion specification 7](#_Toc85529174)

[3.1 Data Mapping Layout 7](#_Toc85529175)

[3.2 Business Rules 7](#_Toc85529176)

[3.3 Data Criteria Specifications 7](#_Toc85529177)

[ Data Selection and Filtering Criteria 7](#_Toc85529178)

[ Pre-Extract Clean up Criteria 7](#_Toc85529179)

[ Post Extract Clean up Criteria 7](#_Toc85529180)

[ Post Extract Acceptance Criteria 8](#_Toc85529181)

[ Post Conversion Acceptance Criteria 8](#_Toc85529182)

[3.4 Reporting Requirements 8](#_Toc85529183)

[3.5 Security & Integrity 9](#_Toc85529184)

[4 Technical Design Specifications 10](#_Toc85529185)

[4.1 Overview 10](#_Toc85529186)

[4.2 Key Logic 10](#_Toc85529187)

[4.3 Data Assembly Characteristics 10](#_Toc85529188)

[5 Conversion Validation 11](#_Toc85529189)

[6 Open Items 12](#_Toc85529190)

[7 Appendix 13](#_Toc85529191)

# Introduction

## Objective

UCPath is the system of record for Employee Data. Employee information is required to perform various functions, for example – approvals in PTP and RTR, Collections in OTC, User accounts for security, etc. This document focuses on providing information required to be extracted from UCPath (UCD HR source of truth) and upload corresponding information in Oracle Cloud.[Source- UCPath and Target : Oracle Cloud HCM]

## Dependencies and Prerequisites

#### Dependencies

1. Employee records in Oracle: All teams – Approvals; OTC – Collectors
2. Default Expense Account assignment for PTP requisition processing
3. PPM: Job Assignment. UCD is not using Oracle Jobs however system expects a value to be populated for every record. (Primary position number will be extracted).

#### Application Configuration Prerequisites

1. Chart of Accounts Segment Values
2. Foundational configuration – COA values, legal entity and HR Organization

# Conversion Description

## Conversion Scope

1. During conversion UCPath will extract all the **active** employees, including student employees, based on effective date igreater than or equal to 30-May-2023 then corresponding record should be extracted and converted to Oracle Cloud. (Volume: 30 to 40k).
2. Employees after May 30, 2023 will be included as part of the employee integration.
3. All PI’s that have an active award will need to be part of the employee extract.
4. Inactive PI’s will not be included in the employee extract.
5. Active contingent workers should be included in the employee extract where, Job Code starts with CWR), and will be determined prior to the extract.
6. Person of interest will **not** be extracted.
7. DVMED employees will be included in the employee extract
8. Employees who are on a short break are considered active

## Assumptions

1. UCD Technical team will extract employee information from UCPath system.

## Proposed Process & Ownership

**Data Extract:**   **Owner: UCD Technical Team**

* UCD Technical team will extract data from source system based on the criteria mentioned in section 3.3.1
* Data extract during each mock cycle will be incremental. Suggestive data volume for each mock conversion is as below:
  + Mock 1 (SIT 1) Target - 75% of Active Employees
  + Mock 2 (SIT 2) Target - 100% of Active Employees
  + Mock 3 (UAT) -100% of Active Employees
  + Cutover - 100% of Active Awards

**Data Transformation:**   **Owner: UCD Technical Team**

* Once the data extract is provided by UCD technical team, UCD technical team will review and advice business to make the necessary changes to the file template
* UCD technical team will provide initial input on the required default values in the template
* Deloitte functional team will validate the data in the template. If there are errors that Deloitte functional can identify prior to load, UCD technical will correct them.

**Data Load Owner: Deloitte Technical Team**

* Deloitte technical team will upload the data into Oracle Cloud using file populated with transformed data
* Once the load is complete, Deloitte technical team will extract a list of error transactions (if any) that may arise due to incorrect formatting, invalid field values, etc. Deloitte will provide guidance to UCD on what the errors signify and how to correct.
* Added 07/25:Post employee conversion the user accounts are created and linked to the employee record and role :UCD Employee JR is assigned. Role code: UCD\_ITE\_EMPLOYEE\_JR
* After loading the Employees, Deloitte technical team will run custom Employee Conversion Reconciliation report to pull the posted data out of Employees tables.
* Deloitte technical team will compare the source system extract data with the custom Employee Conversion Reconciliation reports output from Oracle Cloud Employees; and will share the results with Deloitte functional (for initial review) and UCM business team. Custom report format is defined in section 6.1

**Data Validation & Reconciliation:**   **Owner: UCD Functional**

* As a part of reconciliation process Deloitte functional team will share the **Custom reconciliation report** defined in section 6.1 with UCM functional team.
* UCM functional team will perform a data reconciliation for Mock 1,2,3 and Production by using reconciliation report extract.
* Proposed Validation and Reconciliation timings for each Conversion Cycle
  + Mock 1:
    - Employees – Validate and Reconcile activity (3 days)
  + Mock 2:
    - Employees – Validate and Reconcile activity (3 days)
  + Mock 3:
    - Employees – Validate and Reconcile activity (3 days)
* UCD Functional will validate the reconciliation results and identify any deviations
* Deviations will be handled by UCD Technical
* Final sign off will be provided by UCD Business owners based on the reconciliation results during each cycle.
* Reconciliation results from each conversion cycle will be archived on Box per section 6.1

Roles and Responsibilities for the conversion process are summarized below. Primary owners are responsible for completion of the activity, and secondary owners are identified as parties who may be required to support or provide guidance on the activity.

|  |  |  |
| --- | --- | --- |
| **Activity** | **Primary Owner** | **Secondary Owner** |
| Extract active employee data from UC Path for Conversion | UCD Technical | N/A |
| Review and signoff Employee Conversion input file | UCD Functional | N/A |
| Perform Employee Conversion Load using Integration Program | UCD Technical | N/A |
| Added 0n 07/25: Create user accounts and link it to the employee record and assign role :UCD Employee JR . Role code: UCD\_ITE\_EMPLOYEE\_JR | UCD Technical | N/A |
| Generate Reconciliation report from Oracle | Deloitte Technical | N/A |
| Review reconciliation reports and perform Reconciliation | UCD Functional | Deloitte Functional (during Test cycles) |

# Conversion specification

## Data Mapping Layout

## Business Rules

1. Country Code – If Source system is not providing this value, then default “US”
2. Personal Name – First Name / Last Name /Middle Name – Do not reject any record where source record does not have first name or last name, as per regulations there can be individuals without last name or first name or they may choose not to disclose. From code perspective, object should default FirstNameNotAvailable or LastNameNotAvailable if either of the values is not available in source file.
3. Preferred Name: will be the First Name from the employee record
4. Section Assignment- field Business Unit short name, Code should be either “UCD Business Unit' or “UCD CGA Business Unit”
5. Logic for deriving default expense account for coding <Important, this is not mandatory however important for procure to pay requisition process>

* Technical program should create fields for UCPath to provide values for each segment of COA.
* If source provide values then code should transform old COA to new COA and use it.
* If source do not provide a value, then review value defined in the Integration Framework “expense.account.Non Department”.This will hold default string.
* If any information is not available in source file, then default segment from [combination.](mailto:combination.@Gray)

1. Refer Mapping section for detailed values.
2. Job Code – Default a value <UCD Job> to the field.

Added on 07/25:

1. Post employee conversion the user accounts are created and linked to the employee record and UCD Employee JR is assigned. Role code “UCD\_ITE\_EMPLOYEE\_JR”.

## Data Criteria Specifications

## Data Selection and Filtering Criteria

For conversion previous execution may not exist, to accommodate first execution technical code should extract all records that are active(short workbreak) as on 30-May-2023. Recently terminated employees – should it be extracted?. Not needed.(11/5).

## Pre-Extract Clean up Criteria

* UCD to identify and provide active Employees.
* UCD to provide all Employee related data.
* No Special characters should be part of extracted data other than Name fields. Oracle does accept Hyphenated/apostrophe names.
* There should be no blanks for mandatory columns.

## Post Extract Clean up Criteria

* All the extracted data shall be cleaned up to match the format with Oracle file columns.
* Data formatting will be checked to be in sync with the requirements of the Oracle ERP Cloud (target system).
* Missing required data for non-default fields must be populated by UCD.
* Apply all business rules defined in 3.2

## Post Extract Acceptance Criteria

* Check whether all mandatory information have been populated.
* Ensure the date format in all date fields is in sync with the requirements of the Oracle ERP Cloud

## Post Conversion Acceptance Criteria

* UCD Functional team will validate data with the Custom reconciliation report and provide approval on acceptance on converted data during each cycle.
* For mock conversion cycles, records in error are identified and either corrected and re-loaded during that cycle or accepted and postponed for correction to the next conversion cycle based on UCD business approval, who will decide the criticality and impact.
* For production conversion cycle, records in error are identified and corrected until 100% of the data is successfully loaded.

## Reporting Requirements

**Conversion**

Post every conversion, technical team should provide details of data uploaded in the following format:

1.Statistics based

|  |  |  |
| --- | --- | --- |
|  | Counts | Errors with Details |
| Data Extracted |  |  |
| Data Validated |  |  |
| Data Transformed |  |  |
| Data Uploaded to Oracle |  |  |

2.Data based

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Include Data From | Employee Name  <Last Name> || <First Name>||**Preferred Name** | Employee Email Address | Legal Entity | Cost Center or HR Financial Unit | Employee Default Expense Account | Job Code | Employee Start Date | Employee Effective End Date | Assignment Business Unit | Status(Active/Inactive) | Type(Employee/Contingent) |
| Extract | Yes | Yes | Yes | Yes | **Yes** | **Yes** | Yes | No | No | Yes | Yes |
| Uploaded | Yes | Yes | Yes | Yes | Yes | Yes | Yes | No | Yes | Yes | Yes |

## Security & Integrity

Underline data MUST NOT extract Social security information. Data should not be stored on any Deloitte Laptop and xls extract/file (if any) should be passport et all time.

# Technical Sections

## Source System Data Cleansing

None

## Data Extraction

#### List of Baseline Tables

|  |  |  |
| --- | --- | --- |
| **Database** | **Schema** | **Table Name** |
| AIT\_INT | UCPATH | UC\_JOB\_PRIMARY\_CURRENT |
| AIT\_INT | UCPATH | UC\_PERSON\_PRIMARY\_INFO |
| AIT\_INT | UCPATH | PS\_PER\_ORG\_INST |
| AIT\_INT | UCPATH | PS\_PERSONAL\_PHONE |
| AIT\_INT | UCPATH | UC\_ADDRESSES |
| AIT\_INT | UCPATH | PS\_UC\_WS\_AWARD |
| AIT\_INT | UCPATH | PS\_EMAIL\_ADDRESSES |

#### Data Files

None

#### Staging Tables

* WORKER

|  |  |  |
| --- | --- | --- |
| **Column Name** | **Type** | **Length** |
| META\_DATA | VARCHAR2 | 10 |
| WORKER | VARCHAR2 | 10 |
| SOURCE\_SYSTEM\_OWNER | VARCHAR2 | 10 |
| SOURCE\_SYSTEM\_ID | VARCHAR2 | 20 |
| EFFECTIVE\_START\_DATE | VARCHAR2 | 10 |
| EFFECTIVE\_END\_DATE | VARCHAR2 | 10 |
| PERSON\_NUMBER | VARCHAR2 | 20 |
| START\_DATE | VARCHAR2 | 10 |
| ACTION\_CODE | VARCHAR2 | 20 |

* PERSON\_EMAIL

|  |  |  |
| --- | --- | --- |
| **Column Name** | **Type** | **Length** |
| META\_DATA | VARCHAR2 | 10 |
| PERSON\_EMAIL | VARCHAR2 | 20 |
| SOURCE\_SYSTEM\_OWNER | VARCHAR2 | 10 |
| SOURCE\_SYSTEM\_ID | VARCHAR2 | 20 |
| EMAIL\_TYPE | VARCHAR2 | 10 |
| EMAIL\_ADDRESS | VARCHAR2 | 50 |
| PRIMARY\_FLAG | VARCHAR2 | 1 |

* PERSON\_NAME

|  |  |  |
| --- | --- | --- |
| **Column Name** | **Type** | **Length** |
| META\_DATA | VARCHAR2 | 10 |
| PERSON\_NAME | VARCHAR2 | 20 |
| SOURCE\_SYSTEM\_OWNER | VARCHAR2 | 10 |
| SOURCE\_SYSTEM\_ID | VARCHAR2 | 20 |
| EFFECTIVE\_START\_DATE | VARCHAR2 | 10 |
| EFFECTIVE\_END\_DATE | VARCHAR2 | 10 |
| NAME\_TYPE | VARCHAR2 | 20 |
| LEGISLATION\_CODE | VARCHAR2 | 20 |
| LAST\_NAME | VARCHAR2 | 50 |
| FIRST\_NAME | VARCHAR2 | 50 |
| MIDDLE\_NAME | VARCHAR2 | 50 |
| SUFFIX | VARCHAR2 | 20 |
| KNOWN\_AS | VARCHAR2 | 20 |

* PERSON\_USER\_INFORMATION

|  |  |  |
| --- | --- | --- |
| **Column Name** | **Type** | **Length** |
| META\_DATA | VARCHAR2 | 10 |
| PERSON\_USER\_INFORMATION | VARCHAR2 | 30 |
| SOURCE\_SYSTEM\_OWNER | VARCHAR2 | 10 |
| SOURCE\_SYSTEM\_ID | VARCHAR2 | 20 |
| START\_DATE | VARCHAR2 | 10 |
| USER\_NAME | VARCHAR2 | 20 |

* WORK\_RELATIONSHIP

|  |  |  |
| --- | --- | --- |
| **Column Name** | **Type** | **Length** |
| META\_DATA | VARCHAR2 | 10 |
| WORK\_RELATIONSHIP | VARCHAR2 | 20 |
| SOURCE\_SYSTEM\_OWNER | VARCHAR2 | 10 |
| SOURCE\_SYSTEM\_ID | VARCHAR2 | 20 |
| WORK\_TYPE | VARCHAR2 | 20 |
| PRIMARY\_FLAG | VARCHAR2 | 1 |
| DATE\_START | VARCHAR2 | 10 |
| NEW\_DATE\_START | VARCHAR2 | 10 |
| ACTUAL\_TERM\_DATE | VARCHAR2 | 10 |
| REVOKE\_USER\_ACCESS | VARCHAR2 | 20 |
| REASON\_CODE | VARCHAR2 | 20 |

* WORK\_TERMS

|  |  |  |
| --- | --- | --- |
| **Column Name** | **Type** | **Length** |
| META\_DATA | VARCHAR2 | 10 |
| WORK\_TERMS | VARCHAR2 | 20 |
| SOURCE\_SYSTEM\_OWNER | VARCHAR2 | 10 |
| SOURCE\_SYSTEM\_ID | VARCHAR2 | 20 |
| ACTION\_CODE | VARCHAR2 | 20 |
| REASON\_CODE | VARCHAR2 | 20 |
| EFFECTIVE\_START\_DATE | VARCHAR2 | 10 |
| EFFECTIVE\_END\_DATE | VARCHAR2 | 10 |
| PRIMARY\_WORK\_TERMS\_FLAG | VARCHAR2 | 1 |
| POSITION\_OVERRIDE\_FLAG | VARCHAR2 | 1 |
| EFFECTIVE\_SEQUENCE | VARCHAR2 | 20 |
| EFFECTIVE\_LATEST\_CHANGE | VARCHAR2 | 20 |
| ASSIGNMENT\_STATUS\_TYPE\_CD | VARCHAR2 | 50 |
| ASSIGNMENT\_TYPE | VARCHAR2 | 50 |
| BUSINESS\_UNIT | VARCHAR2 | 20 |
| LEGAL\_EMPLOYER\_NM | VARCHAR2 | 50 |

* ASSIGNMENT

|  |  |  |
| --- | --- | --- |
| **Column Name** | **Type** | **Length** |
| META\_DATA | VARCHAR2 | 10 |
| ASSIGNMENT | VARCHAR2 | 20 |
| SOURCE\_SYSTEM\_OWNER | VARCHAR2 | 10 |
| SOURCE\_SYSTEM\_ID | VARCHAR2 | 20 |
| EFFECTIVE\_START\_DATE | VARCHAR2 | 10 |
| EFFECTIVE\_END\_DATE | VARCHAR2 | 10 |
| EFFECTIVE\_SEQUENCE | VARCHAR2 | 20 |
| JOB\_CODE | VARCHAR2 | 20 |
| DEPARTMENT\_NAME | VARCHAR2 | 30 |
| LOCATION\_CODE | VARCHAR2 | 20 |
| ASSIGNMENT\_STATUS\_TYPE\_CODE | VARCHAR2 | 20 |
| PRIMARY\_ASSIGN\_FLAG | VARCHAR2 | 1 |
| ASSIGNMENT\_TYPE | VARCHAR2 | 20 |
| DEFAULTEXPENSEACCOUNT | VARCHAR2 | 100 |

* PERSON\_PHONE

|  |  |  |
| --- | --- | --- |
| **Column Name** | **Type** | **Length** |
| META\_DATA | VARCHAR2 | 10 |
| PERSON\_PHONE | VARCHAR2 | 20 |
| SOURCE\_SYSTEM\_OWNER | VARCHAR2 | 10 |
| SOURCE\_SYSTEM\_ID | VARCHAR2 | 20 |
| PHONE\_TYPE | VARCHAR2 | 10 |
| PHONE\_NUMBER | VARCHAR2 | 30 |

* PERSON\_ADDRESS

|  |  |  |
| --- | --- | --- |
| **Column Name** | **Type** | **Length** |
| META\_DATA | VARCHAR2 | 10 |
| PERSON\_ADDRESS | VARCHAR2 | 20 |
| SOURCE\_SYSTEM\_OWNER | VARCHAR2 | 10 |
| SOURCE\_SYSTEM\_ID | VARCHAR2 | 20 |
| ADDRESS\_TYPE | VARCHAR2 | 20 |
| ADDRESS\_LINE1 | VARCHAR2 | 40 |
| CITY | VARCHAR2 | 40 |
| POSTAL\_CD | VARCHAR2 | 20 |
| EFFECTIVE\_START\_DT | VARCHAR2 | 10 |
| EFFECTIVE\_END\_DT | VARCHAR2 | 10 |

#### Query to Populate Staging Tables

insert into WORKER (

META\_DATA

, WORKER

, SOURCE\_SYSTEM\_OWNER

, SOURCE\_SYSTEM\_ID

, EFFECTIVE\_START\_DATE

, EFFECTIVE\_END\_DATE

, PERSON\_NUMBER

, START\_DATE

, ACTION\_CODE

)

select

'MERGE' as META\_DATA

, 'Worker' as WORKER

, 'UCPATH' as SOURCE\_SYSTEM\_OWNER

, jb.EMPLID as SOURCE\_SYSTEM\_ID

, to\_char(jb.ASGN\_START\_DT, 'YYYY/MM/DD') AS EFFECTIVE\_START\_DATE

, '4712/12/31' AS EFFECTIVE\_END\_DATE

, jb.EMPLID AS PERSON\_NUMBER

, to\_char(jb.HIRE\_DT, 'YYYY/MM/DD') as START\_DATE

, 'HIRE' AS ACTION\_CODE

FROM UC\_JOB\_PRIMARY\_CURRENT jb

WHERE jb.EMPL\_STATUS IN ('A','P','L','W') --INCLUDE ACTIVE EMPLOYEES - Employees who are on a short break are considered active

AND jb.POI\_TYPE = ' ' --EXCLUDE PERSONS OF INTEREST

;

insert into PERSON\_EMAIL (

META\_DATA

, PERSON\_EMAIL

, SOURCE\_SYSTEM\_OWNER

, SOURCE\_SYSTEM\_ID

, PERSON\_ID

, DATE\_FROM

, EMAIL\_TYPE

, EMAIL\_ADDRESS

, PRIMARY\_FLAG

)

SELECT

'MERGE' as META\_DATA

, 'PersonEmail' as PERSON\_EMAIL

, 'UCPATH' as SOURCE\_SYSTEM\_OWNER

, jb.EMPLID || '\_PRSN\_EMAIL' as SOURCE\_SYSTEM\_ID

, jb.EMPLID as PERSON\_ID

, to\_char(jb.hire\_dt, 'YYYY/MM/DD') as DATE\_FROM

, 'W1' as EMAIL\_TYPE

, email.EMAIL\_ADDR as EMAIL\_ADDRESS

, 'Y' as PRIMARY\_FLAG

FROM UC\_JOB\_PRIMARY\_CURRENT jb

INNER JOIN PS\_EMAIL\_ADDRESSES email ON jb.EMPLID = email.EMPLID

WHERE jb.EMPL\_STATUS IN ('A','P','L','W') -- INCLUDE ACTIVE EMPLOYEES - Employees who are on a short break are considered active

AND jb.POI\_TYPE = ' ' -- EXCLUDE PERSONS OF INTEREST

AND email.E\_ADDR\_TYPE = 'BUSN'

;

INSERT INTO PERSON\_NAME (

META\_DATA

, PERSON\_NAME

, SOURCE\_SYSTEM\_OWNER

, SOURCE\_SYSTEM\_ID

, PERSON\_ID

, EFFECTIVE\_START\_DATE

, EFFECTIVE\_END\_DATE

, NAME\_TYPE

, LEGISLATION\_CODE

, LAST\_NAME

, FIRST\_NAME

, MIDDLE\_NAMES

, SUFFIX

, KNOWN\_AS

)

SELECT

'MERGE' as META\_DATA

, 'PersonName' as PERSON\_NAME

, 'UCPATH' as SOURCE\_SYSTEM\_OWNER

, jb.EMPLID || '\_GLOBAL' as SOURCE\_SYSTEM\_ID

, jb.EMPLID as PERSON\_ID

, to\_char(jb.ASGN\_START\_DT, 'YYYY/MM/DD') as EFFECTIVE\_START\_DATE

, '4712/12/31' as EFFECTIVE\_END\_DATE

, 'GLOBAL' as NAME\_TYPE

, 'US' as LEGISLATION\_CODE

, substr(coalesce(psn.LAST\_NAME, 'UNKNOWN'), 1, 50) as LAST\_NAME

, substr(coalesce(psn.FIRST\_NAME, 'UNKNOWN'), 1, 50) as FIRST\_NAME

, substr(psn.MIDDLE\_NAME, 1, 50) as MIDDLE\_NAMES

, psn.PRIMARY\_NAME\_SUFFIX as SUFFIX

, substr(coalesce(psn.PRIMARY\_FIRST\_NAME, 'UNKNOWN'), 1, 20) as KNOWN\_AS

FROM UC\_JOB\_PRIMARY\_CURRENT jb

LEFT OUTER JOIN UC\_PERSON\_PRIMARY\_INFO psn ON psn.EMPLID = jb.EMPLID

WHERE jb.EMPL\_STATUS IN ('A','P','L','W') -- INCLUDE ACTIVE EMPLOYEES - Employees who are on a short break are considered active

AND jb.POI\_TYPE = ' ' -- EXCLUDE PERSONS OF INTEREST

;

INSERT INTO PERSON\_USER\_INFORMATION (

META\_DATA

, PERSON\_USER\_INFORMATION

, SOURCE\_SYSTEM\_OWNER

, SOURCE\_SYSTEM\_ID

, PERSON\_ID

, PERSON\_NUMBER

, START\_DATE

, USER\_NAME

)

SELECT

'MERGE' as META\_DATA

, 'PersonUserInformation' as PERSON\_USER\_INFORMATION

, 'UCPATH' as SOURCE\_SYSTEM\_OWNER

, jb.EMPLID || '\_USINF' as SOURCE\_SYSTEM\_ID

, jb.EMPLID as PERSON\_ID

, jb.EMPLID as PERSON\_NUMBER

, to\_char(jb.hire\_dt, 'YYYY/MM/DD') as START\_DATE

, person.COMP\_ACCT\_USER\_ID as USER\_NAME

FROM UC\_JOB\_PRIMARY\_CURRENT jb

INNER JOIN UC\_KRIM\_PERSON\_MV person on jb.EMPLID = person.EMPLOYEE\_ID

WHERE jb.EMPL\_STATUS IN ('A','P','L','W') -- INCLUDE ACTIVE EMPLOYEES - Employees who are on a short break are considered active

AND jb.POI\_TYPE = ' ' -- EXCLUDE PERSONS OF INTEREST

;

INSERT INTO WORK\_RELATIONSHIP (

META\_DATA

, WORK\_RELATIONSHIP

, SOURCE\_SYSTEM\_OWNER

, SOURCE\_SYSTEM\_ID

, PERSON\_ID

, WORKER\_TYPE

, PRIMARY\_FLAG

, DATE\_START

, NEW\_DATE\_START

, ACTUAL\_TERMINATION\_DATE

, REVOKE\_USER\_ACCESS

, REASON\_CODE

, LEGAL\_EMPLOYER\_NAME

)

SELECT

'MERGE' as META\_DATA

, 'WorkRelationship' as WORK\_RELATIONSHIP

, 'UCPATH' as SOURCE\_SYSTEM\_OWNER

, jb.EMPLID || '\_POS' as SOURCE\_SYSTEM\_ID

, jb.EMPLID as PERSON\_ID

, decode(jb.PER\_ORG, 'CWR', 'C', 'E') as WORKER\_TYPE

, 'Y' as PRIMARY\_FLAG

, to\_char(jb.hire\_dt, 'YYYY/MM/DD') as DATE\_START

, to\_char(jb.last\_hire\_dt, 'YYYY/MM/DD') as NEW\_DATE\_START

, to\_char(jb.termination\_dt, 'YYYY/MM/DD') as ACTUAL\_TERMINATION\_DATE

, '' as REVOKE\_USER\_ACCESS

, CASE

WHEN jb.PER\_ORG = 'EMP' THEN 'HIRE'

WHEN jb.PER\_ORG = 'CWR' THEN 'PLACE'

ELSE ''

END as REASON\_CODE

, decode(jb.business\_unit, 'DVCMP', 'UC Davis - Excluding Schools of Health', 'DVMED', 'UC Davis Schools of Health', 'UCANR', 'UC Davis - Excluding Schools of Health', 'UC Davis - Excluding Schools of Health') as LEGAL\_EMPLOYER\_NAME

FROM UC\_JOB\_PRIMARY\_CURRENT jb

WHERE jb.EMPL\_STATUS IN ('A','P','L','W') -- INCLUDE ACTIVE EMPLOYEES - Employees who are on a short break are considered active

AND jb.POI\_TYPE = ' ' -- EXCLUDE PERSONS OF INTEREST

;

INSERT INTO WORK\_TERMS (

META\_DATA

, WORK\_TERMS

, SOURCE\_SYSTEM\_OWNER

, SOURCE\_SYSTEM\_ID

, PERSON\_ID

, ACTION\_CODE

, REASON\_CODE

, EFFECTIVE\_START\_DATE

, EFFECTIVE\_END\_DATE

, PRIMARY\_WORK\_TERMS\_FLAG

, POSITION\_OVERRIDE\_FLAG

, EFFECTIVE\_SEQUENCE

, EFFECTIVE\_LATEST\_CHANGE

, ASSIGNMENT\_STATUS\_TYPE\_CODE

, ASSIGNMENT\_TYPE

, BUSINESS\_UNIT\_SHORT\_CODE

, LEGAL\_EMPLOYER\_NM

, PERIOD\_OF\_SERVICE\_ID

)

SELECT

'MERGE' as META\_DATA

, 'WorkTerms' as WORK\_TERMS

, 'UCPATH' as SOURCE\_SYSTEM\_OWNER

, jb.EMPLID || '\_EMP\_TERM' as SOURCE\_SYSTEM\_ID

, jb.EMPLID as PERSON\_ID

, 'HIRE' as ACTION\_CODE

, CASE WHEN jb.PER\_ORG = 'CWR' THEN 'PLACE'

WHEN jb.PER\_ORG = 'EMP' THEN 'HIRE'

ELSE ''

END as REASON\_CODE

, to\_char(jb.ASGN\_START\_DT, 'YYYY/MM/DD') as EFFECTIVE\_START\_DATE

, '4712/12/31' as EFFECTIVE\_END\_DATE

, 'Y' as PRIMARY\_WORK\_TERMS\_FLAG

, 'N' as POSITION\_OVERRIDE\_FLAG

, '1' as EFFECTIVE\_SEQUENCE

, 'Y' as EFFECTIVE\_LATEST\_CHANGE

, decode(jb.HR\_STATUS, 'A', 'ACTIVE\_PROCESS', 'I', 'INACTIVE\_PROCESS') as ASSIGNMENT\_STATUS\_TYPE\_CODE

, decode(jb.PER\_ORG, 'CWR', 'CT', 'ET') as ASSIGNMENT\_TYPE

, 'UCD Business Unit' as BUSINESS\_UNIT\_SHORT\_CODE

, decode(jb.business\_unit, 'DVCMP', 'UC Davis - Excluding Schools of Health', 'DVMED', 'UC Davis Schools of Health', 'UCANR', 'UC Davis - Excluding Schools of Health', 'UC Davis - Excluding Schools of Health') as LEGAL\_EMPLOYER\_NAME

, jb.EMPLID || '\_POS' as PERIOD\_OF\_SERVICE\_ID -- same value as SourceSystemId from WorkRelationships above

FROM UC\_JOB\_PRIMARY\_CURRENT jb

WHERE jb.EMPL\_STATUS IN ('A','P','L','W') -- INCLUDE ACTIVE EMPLOYEES - Employees who are on a short break are considered active

AND jb.POI\_TYPE = ' ' -- EXCLUDE PERSONS OF INTEREST

;

INSERT INTO ASSIGNMENT (

META\_DATA

, ASSIGNMENT

, SOURCE\_SYSTEM\_OWNER

, SOURCE\_SYSTEM\_ID

, PERSON\_ID

, WORK\_TERMS\_ASSIGNMENT\_ID

, JOB\_CODE

, DEPARTMENT\_NAME

, ASSIGNMENT\_STATUS\_TYPE\_CODE

, LOCATION\_CODE

, DEFAULT\_EXPENSE\_ACCOUNT

, BUSINESS\_UNIT\_SHORT\_CODE

, EFFECTIVE\_SEQUENCE

, EFFECTIVE\_START\_DATE

, EFFECTIVE\_END\_DATE

, PRIMARY\_ASSIGNMENT\_FLAG

, ASSIGNMENT\_TYPE

, ACTION\_CODE

, EFFECTIVE\_LATEST\_CHANGE

)

SELECT

'MERGE' as META\_DATA

, 'Assignment' as ASSIGNMENT

, 'UCPATH' as SOURCE\_SYSTEM\_OWNER

, jb.EMPLID || '\_EMP\_ASG' as SOURCE\_SYSTEM\_ID

, jb.EMPLID as PERSON\_ID

, jb.EMPLID || '\_EMP\_TERM' as WORK\_TERMS\_ASSIGNMENT\_ID -- must be same as SourceSystemId from WorkTerms above

, jb.JOBCODE as JOB\_CODE

--, CASE WHEN d.FIN\_DEPT IS NOT NULL AND d.FIN\_DEPT\_DESC IS NOT NULL

--THEN substr(d.FIN\_DEPT || ' - ' || d.FIN\_DEPT\_DESC, 1, 50)

--ELSE '0100311 - Agr & Env Sci Deans Office' -- DEFAULT FOR NOW, WILL CHANGE

--END as DEPARTMENT\_NAME

, '0100311 - Agr & Env Sci Deans Office' as DEPARTMENT\_NAME -- DEFAULT FOR NOW, WILL CHANGE

, decode(jb.HR\_STATUS, 'A', 'ACTIVE\_PROCESS', 'I', 'INACTIVE\_PROCESS') as ASSIGNMENT\_STATUS\_TYPE\_CODE

, '100' as LOCATION\_CODE

, '' as DEFAULT\_EXPENSE\_ACCOUNT --TODO: default expense account

, 'UCD Business Unit' as BUSINESS\_UNIT\_SHORT\_CODE

, '1' as EFFECTIVE\_SEQUENCE

, to\_char(jb.ASGN\_START\_DT, 'YYYY/MM/DD') as EFFECTIVE\_START\_DATE

, '4712/12/31' as EFFECTIVE\_END\_DATE

, 'Y' as PRIMARY\_ASSIGNMENT\_FLAG

, CASE

WHEN jb.action = 'HIR' and jb.PER\_ORG = 'CWR' THEN 'C'

WHEN jb.action = 'HIR' and jb.PER\_ORG = 'EMP' THEN 'E'

ELSE ''

END as ASSIGNMENT\_TYPE

, 'HIRE' as ACTION\_CODE

, 'Y' as EFFECTIVE\_LATEST\_CHANGE

FROM UC\_JOB\_PRIMARY\_CURRENT jb

LEFT JOIN HR\_EMPLOYEE\_DEPARTMENT\_MAPPING\_BASELINE d on jb.deptid = d.dept\_cd

WHERE jb.EMPL\_STATUS IN ('A','P','L','W') -- INCLUDE ACTIVE EMPLOYEES - Employees who are on a short break are considered active

AND jb.POI\_TYPE = ' ' -- EXCLUDE PERSONS OF INTEREST

;

INSERT INTO PERSON\_PHONE (

META\_DATA

, PERSON\_PHONE

, SOURCE\_SYSTEM\_OWNER

, SOURCE\_SYSTEM\_ID

, PERSON\_ID

, PHONE\_TYPE

, PHONE\_NUMBER

, DATE\_FROM

)

SELECT

'MERGE' as META\_DATA

, 'PersonPhone' as PERSON\_PHONE

, 'UCPATH' as SOURCE\_SYSTEM\_OWNER

, jb.EMPLID || '\_PHONE' as SOURCE\_SYSTEM\_ID

, jb.EMPLID as PERSON\_ID

, 'W1' as PHONE\_TYPE

, '1-503-752-1011' as PHONE\_NUMBER

, to\_char(jb.hire\_dt, 'YYYY/MM/DD') as DATE\_FROM

FROM UC\_JOB\_PRIMARY\_CURRENT jb

WHERE jb.EMPL\_STATUS IN ('A','P','L','W') -- INCLUDE ACTIVE EMPLOYEES - Employees who are on a short break are considered active

AND jb.POI\_TYPE = ' ' -- EXCLUDE PERSONS OF INTEREST

;

INSERT INTO PERSON\_ADDRESS (

META\_DATA

, PERSON\_ADDRESS

, SOURCE\_SYSTEM\_OWNER

, SOURCE\_SYSTEM\_ID

, PERSON\_ID

, ADDRESS\_TYPE

, ADDRESS\_LINE1

, CITY

, STATE

, POSTAL\_CD

, COUNTRY

, EFFECTIVE\_START\_DT

, EFFECTIVE\_END\_DT

)

SELECT

'MERGE' as META\_DATA

, 'PersonAddress' as PERSON\_ADDRESS

, 'UCPATH' as SOURCE\_SYSTEM\_OWNER

, jb.EMPLID || '\_ADDRESS' as SOURCE\_SYSTEM\_ID

, jb.EMPLID as PERSON\_ID

, 'MAIL' as ADDRESS\_TYPE

, 'One Shields Avenue' as ADDRESS\_LINE1

, 'Davis' as CITY

, 'CA' as STATE

, '95616' as POSTAL\_CD

, 'US' as COUNTRY

, to\_char(jb.HIRE\_DT, 'YYYY/MM/DD') as EFFECTIVE\_START\_DT -- must be >= start-date of WORKER

, '4712/12/31' as EFFECTIVE\_END\_DT

FROM UC\_JOB\_PRIMARY\_CURRENT jb

WHERE jb.EMPL\_STATUS IN ('A','P','L','W') -- INCLUDE ACTIVE EMPLOYEES - Employees who are on a short break are considered active

AND jb.POI\_TYPE = ' ' -- EXCLUDE PERSONS OF INTEREST

;

## Data Enrich and Transformation in Staging Tables

* No Special characters should be part of extracted data other than Name fields. Oracle does accept Hyphenated/apostrophe names.
* There should be no blanks for mandatory columns.
* Data formatting will be checked to be in sync with the requirements of the Oracle ERP Cloud (target system).
* Apply all business rules defined in 3.2

## Import Data Files to Tables

None

## Generate Zipped CSV Files

Groovy File Name: generateWorker.groovy

Package: etl/export/jobs/fin

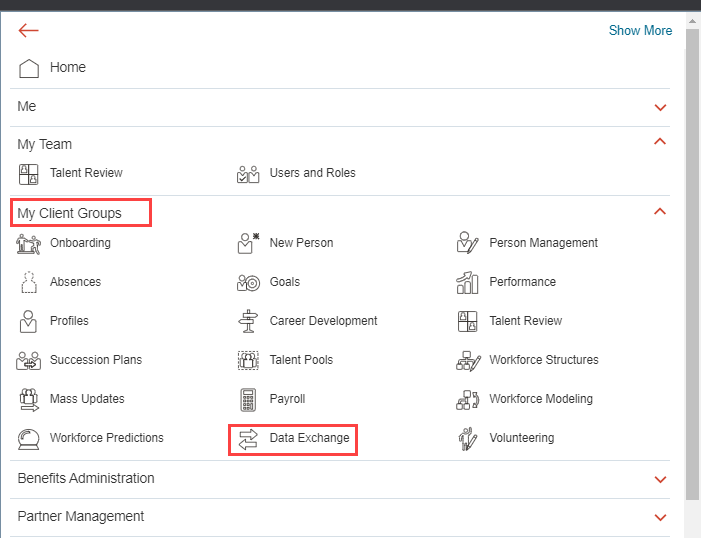
Class Name: generateWorker

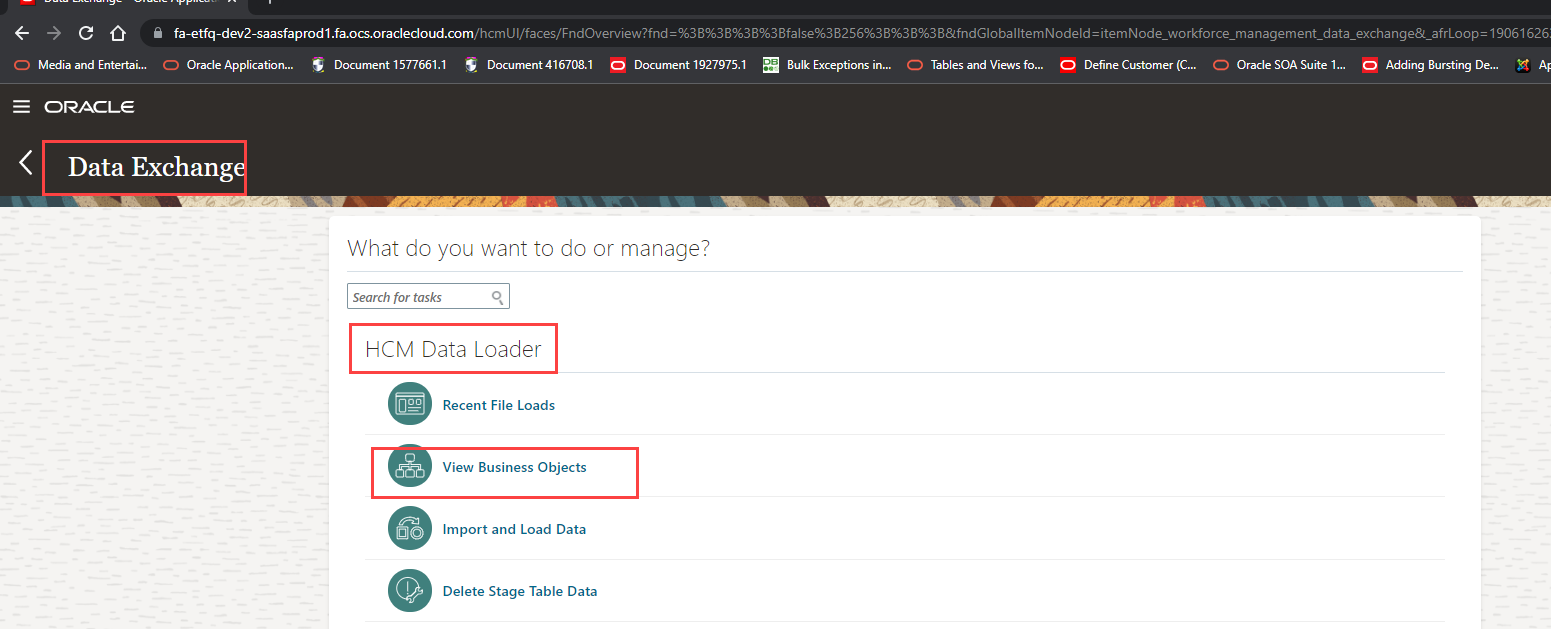
Logic:

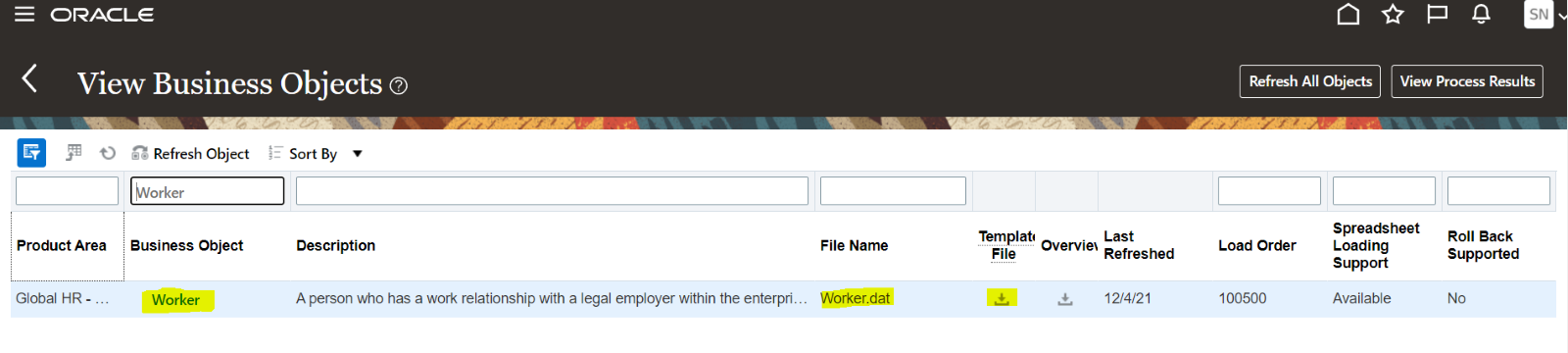
1. Method: populateFileFromTables to generate the Worker.data file.
2. Call util/zipUtil.zip to zip Worker.data into Worker.zip file..

Download template file from below navigation.

Navigator >> Data Exchange >> HCM Data Loader >> View Business Object >> Search for ‘Worker’ business object >> Click download icon for Template File and Overview File.



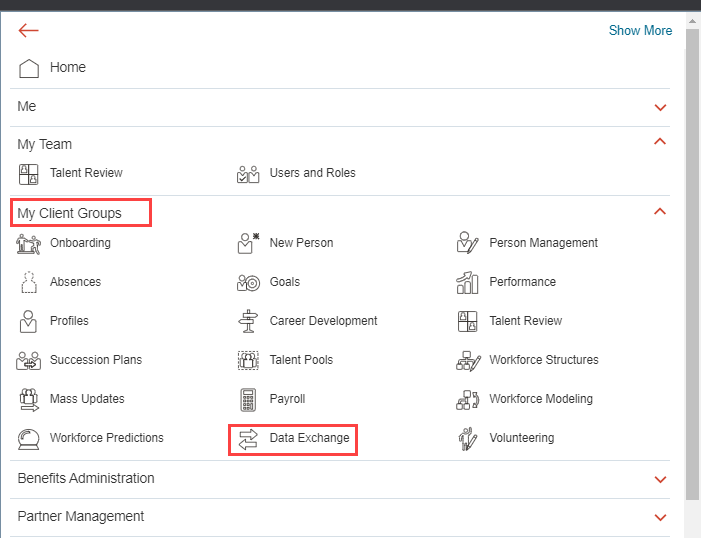


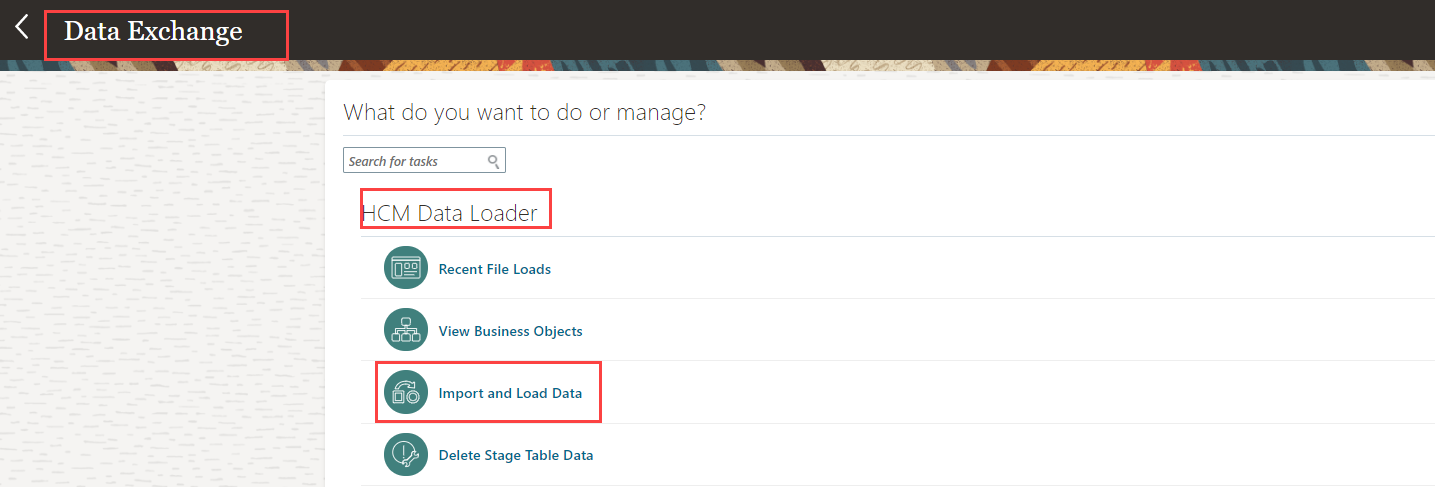
  
**Steps to load data into HCM Data Loader:**

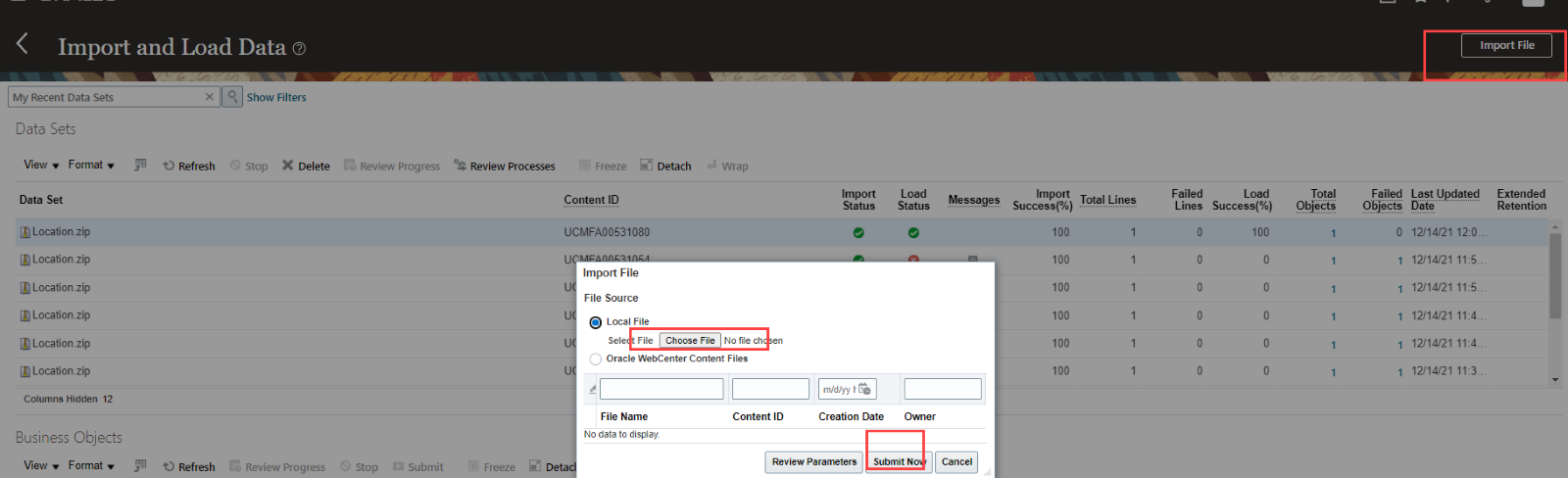
Aligning to above template, map all required fields with a pipe delimited flat file.

Save the file with .dat extension and zip it.

Upload the generated zip file and submit ‘Import’ – Navigator >> Data Exchange >> HCM Data Loader >> Import and Load Data >> Import File >> Upload the File >> Submit Now.







# Conversion Validation

# Open Items

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Date Reported | Status | Question | Answer | Due Date | Responsible Party |
| 10/15/21 | Open | Is the supervisor relationship needed for SCM approvals ? | No. Its position based reporting. |  |  |
| 10/15/21 | Open | Are job codes a requirement for PPM? | Yes |  |  |
| 10/15/21 | Open | Rules of Inactive Employees – how far do we go back? | TBD |  |  |
| 10/15/21 | Open | Will UCD use a location default value as UCM did? | What is the Purpose of location?  Ans: Its only used for Reporting purpose. A separate location conversion/interface is needed if Employee location is needed. |  |  |
| 10/15/21 | Open | Business Unit: is this mandatory? Since source system does not have the BU, need to determine the criteria to derive the BU. |  |  |  |

# Appendix

Appendix A - Acronyms

| Acronym | Literal Translation |
| --- | --- |
| SOA | Service Oriented Architecture |
| SOACS | SOA Cloud Service |
| BPEL | Business Process Execution Language |
| SCA | Service Component Architecture |
| WSDL | Web Service Description Language |
| Cloud ESS | Enterprise Scheduling Service |
| AP | Accounts Payables |
| AR | Accounts Receivables |
| GL | General Ledger |
| MFT | Managed File Transfer |
| FBDI | File Based Data Import |
| JCS | Java Cloud Service |
| JSON | JavaScript Object Notation |
| SFTP | SSH File Transfer Protocol |
| API | Application Programming Interface |
| ESB | Enterprise Service Bus |
| SQS | Simple Queue Service |
| SOAP | Simple Object Access Protocol |
| ADF | Oracle: Application Development Framework |
| EDM | Enterprise Data Management |
| GOA | Go Anywhere Server (UC Berkley) |
| COA | Chart of Accounts |

Appendix B: Relevant Documents

The following functional design documents have been referred while creating this document.

| Document Name | Location / URL |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |