An Oracle White Paper
February 2012

Defining a Simple HR Outbound Interface Using HCM Extracts [ID 1429892.1]

Executive Overview

This white paper is intended to provide technical guidance to implementation teams planning to build their own interface solution utilizing the Fusion HCM Applications, Extracts functionality.

Introduction

The Fusion Applications HCM Extract product allows customers to build custom defined data extracts to export business data. The HCM Extract functionality is highly flexible and configurable, utilizing a multi threaded backend for the extraction process, and a BI Publisher based front end for presenting the end result. The flow of information through the system is tightly integrated resulting in an end result that can be submitted with minimal user intervention.

This white paper provides technical guidance for developers tasked with producing a simple HR data extract suitable for interfacing to an external provider. The information provided here is intended as an introductory guide but will not cover all of the details required or provide any particular interface definition.

This document contains guidance on

- Designing an Extract
- Creating the Extract
- Running the Extract
- Implementing an Incremental Extract

.

Before you start

This document assumes knowledge of the Fusion HCM Human Resources data model, and knowledge of the HCM Extracts feature.

The extract results are delivered using BI Publisher, if you wish to use some of the advanced formatting possible with BI Publisher you will need to understand this product.

Additional information will be provided over the next few months.

Technical Pre Requisites

- The HCM Extracts process requires that a payroll relationship is configured. This is normally done as part of the provisioning process, and is would normally be required if you need to create element entries.

Other Information

- HCM Extract documentation – TBD
- HCM Data Base Item Guidelines – TBD

Glossary

Database Item

A database item is a basic building block that may be used to build an extract, it provides a predefined programmatic approach for retrieving an individual piece of information. Database Items (DBI's) can be grouped together . Behind a database item is a SQL query. The SQL query is defined in a route. A route is connected to a database item by a user entity.

Data Base Item Group

A database item group allows a database item to execute different route code depending upon the context presented to the DBI at runtime. This allows for flexibility in that the same DBI may be used at multiple levels within the employment hierarchy.

User entity

The user entity provides a link between a DBI and the underlying route code.

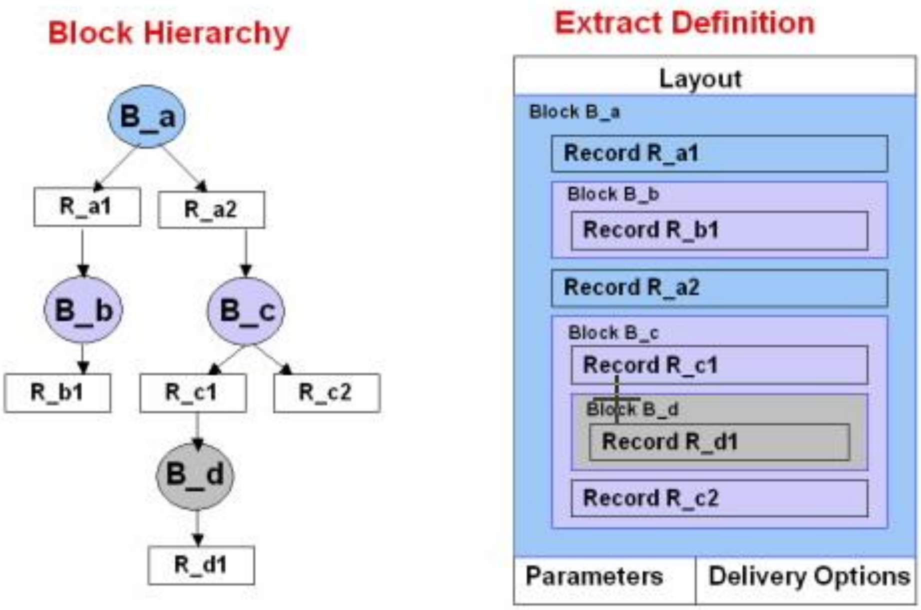
Contexts

A context is essentially a parameter supplied to the route code to ensure that the right level of data is returned. Contexts may be optional, allowing data to be returned as a summary or at a lower level.

Overview of the Extract definition

The HCM Extracts functionality makes use of data base items to retrieve data from the Human Resources product's tables. A list of the available data base items can be found in an accompanying spreadsheet. Refer to the "Other Information" for links.

The basic structure of an extract is



A complete Extract definition will consist of

- Parameters
- Blocks
- Block Links
- Filter Criteria
- Records
- Data Elements
- Conditional Actions and Delivery options

Parameters

Parameters may be defined for each extract definition and these would correspond to ESS (Enterprise Scheduler Service) parameters provided during the execution of the

Extract process. A set of standard parameters will be created automatically upon saving an extract definition. Additional parameters may be specified should your extract require.

Blocks

Fusion HCM Extracts uses Fast Formula User Entity/Route to define the query for obtaining data for a block. Database items (DBIs) and Database Item Groups would be created for these User Entities, which may be used as the basis for Data Elements within a record and/or for setting the values of relevant contexts for the block.

The block essentially defines, by way of a reference to a User Entity, the SQL query/cursor FROM and general WHERE clause, along with columns that can be part of the SELECT clause per the Database Items defined for that User Entity.

Block Links

Block links would be used to form a block hierarchy or sequence, with records in each block defining the actual sequencing of block hierarchy traversal relative to other records in the block.

Block links are based upon a DBI (hence DBI group) defined for each of the respective block user entities, specifying an attribute from each block with which to form the join criteria.   A block may be linked to at most one parent block, but may be linked to that parent block using multiple DBI pairs, hence join conditions.

Block Filters

Block filters provide an additional level of filtering of the data returned within a block. Additional block filter criteria may be applied, either in the form of condition expressions referencing those same Database Items, or to handle more complex cases, in the form of Fast Formula making use of other DBIs/functions that reference the context values set for the block records

The Record Layout is a physical collection of Data Elements. The Block is a grouping of Record layouts. Blocks may be associated with multiple records, and a record may specify the next (child) block to be extracted. This information will define the execution path or traversal used for the data extraction.

Data Elements

Data elements are defined within a record and can derive their data from multiple sources. A data element is used to retrieve a single piece of data. In this example we will use DBI GROUP as the data source for the record. Most HR information can be retrieved through the use of a DBI (Data Base Items).

DBI’s often use context to determine the scope of the information retrieved. The context is an important area to understand for each DBI and can be determined from the DBI Reference spreadsheet.

Designing the Extract Structure

All HR information related to employees can be grouped by Legal Employer, this paper assumes that the Legal employer will be used to drive the extract. Therefore the extract will fetch all employees for a particular legal employer.

Additionally each person will have at least one assignment or terms, this white paper will use the employee's assignment to fetch position, job and additional assignment related information. Multiple records may be produced in the case of multiple employments.

The Extract we discuss here will therefore define two basic levels of data, employee level data which will be by legal employer and work relationship. The second level will be at the assignment level and will fetch by Assignment id and effective date.

The following diagram provides an overview of the structure of this simple extract process.

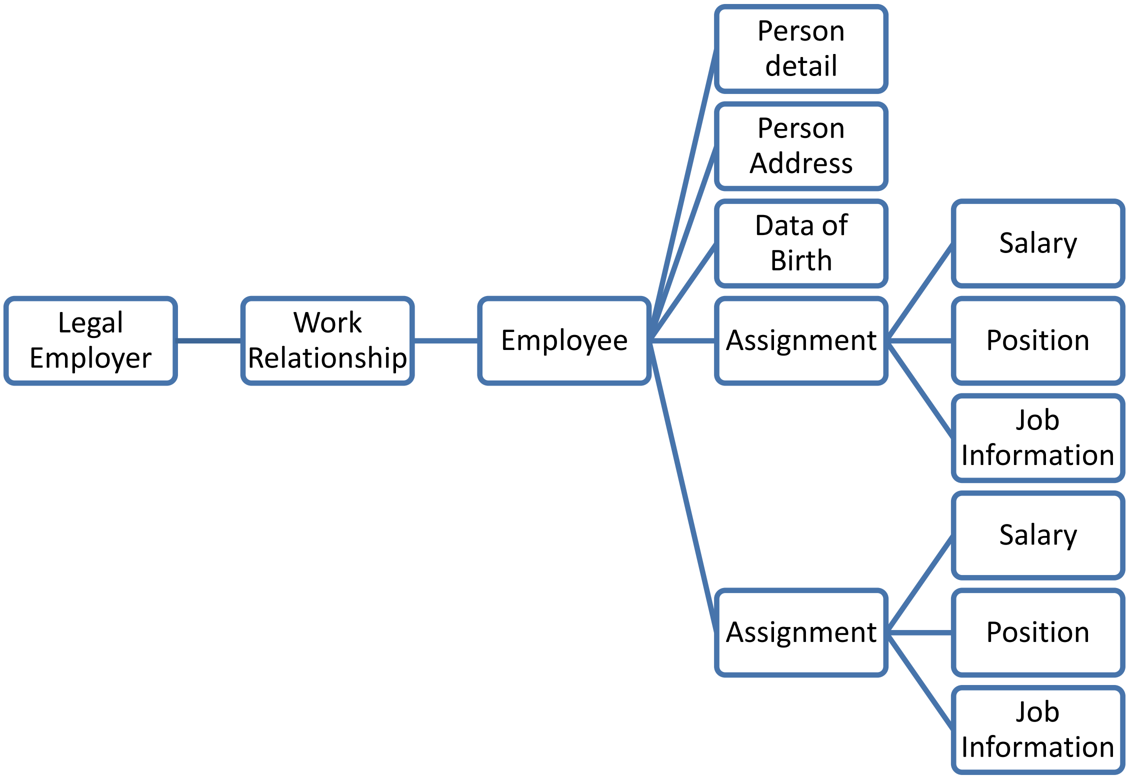


Figure 1. Data model Overview of this simple Extract structure

Creating an Extract Definition

This section will outline the steps required to create a simple extract definition. The basic steps to creating an extract definition are:

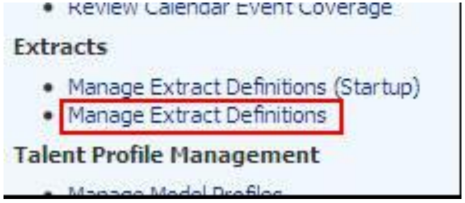
1. Create Extract Definition
2. Define Extract parameters
3. Define Block
4. Define Records
5. Define Elements
6. Configure Block Link

1 – Create Extract Definition

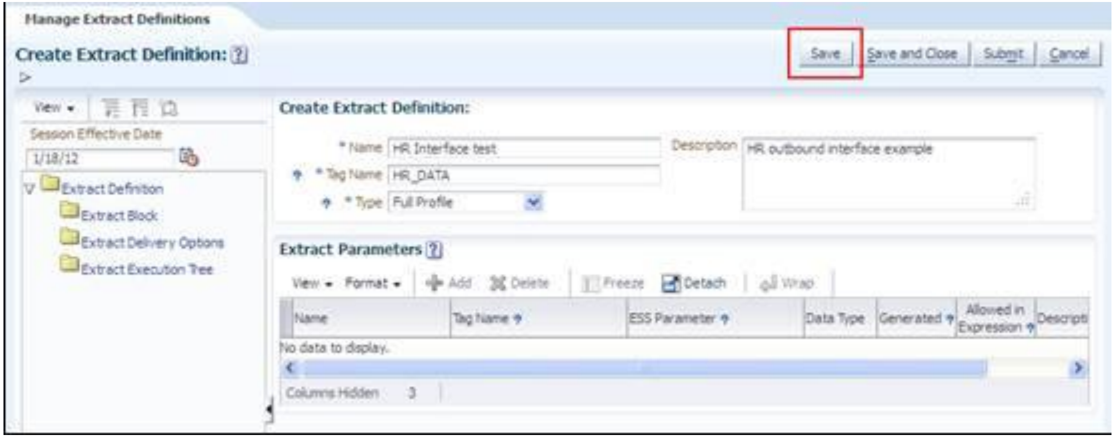
The first step in the process is to define the extract.

a) Login to Hcm Core Setup

b) Extracts-> Manage Extract Definitions.


Review Calendar Event Coverage
Extracts
Manage Extract Definitions (Startup)
Manage Extract Definitions
Talent Profile Management
Manage Model Profiles

c) Now enter basic information about your extract



Create Extract Definition:

Session Effective Date 1/18/12

Create Extract Definition:

* Name HR Interface test

* Tag Name HR_DATA

* Type Full Profile

Description HR outbound interface example

Extract Parameters

Name	Tag Name	ESS Parameter	Data Type	Generated	Allowed in Expression	Descripti
No data to display.						

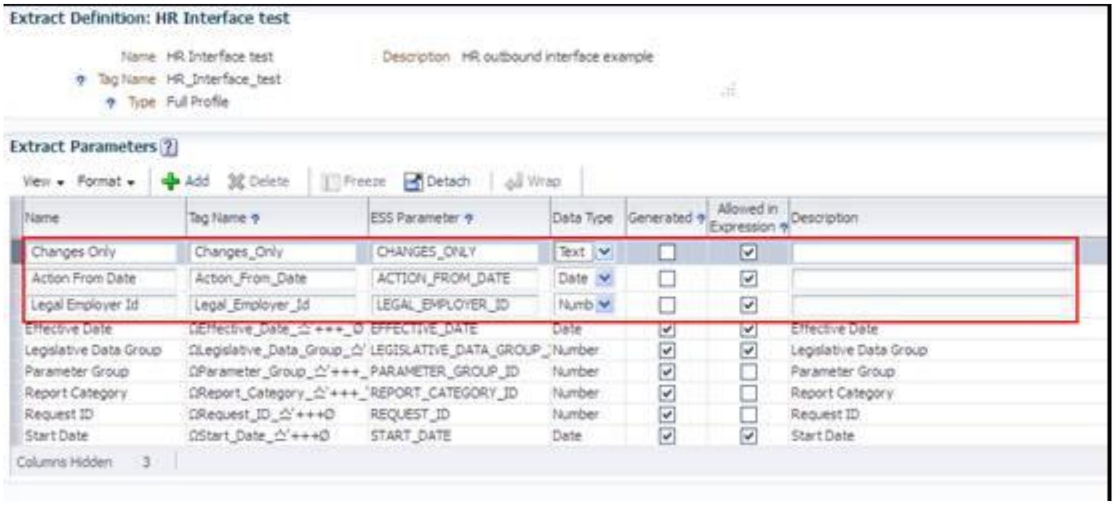
Columns Hidden 3

d) Click Save to complete this step.

2 – Define Extract Parameters

Once the basic definition has been created the next step is to define parameters.

a) Use **Add** to create 3 user parameters, you will also notice that a number of predefined system parameters will be created automatically



Extract Definition: HR Interface test

Name HR Interface test

Tag Name HR_Interface_test

Type Full Profile

Description HR outbound interface example

Extract Parameters

Name	Tag Name	ESS Parameter	Data Type	Generated	Allowed in Expression	Description
Changes Only	Changes_Only	CHANGES_ONLY	Text	□	☑	
Action From Date	Action_From_Date	ACTION_FROM_DATE	Date	□	☑	
Legal Employer Id	Legal_Employer_Id	LEGAL_EMPLOYER_ID	Numb	□	☑	
Effective Date	ΩEffective_Date_△'+++_Ω	EFFECTIVE_DATE	Date	☑	☑	Effective Date
Legislative Data Group	ΩLegislative_Data_Group_△'	LEGISLATIVE_DATA_GROUP_	Number	☑	☑	Legislative Data Group
Parameter Group	ΩParameter_Group_△'+++_	PARAMETER_GROUP_ID	Number	☑	□	Parameter Group
Report Category	ΩReport_Category_△'+++_	REPORT_CATEGORY_ID	Number	☑	□	Report Category
Request ID	ΩRequest_ID_△'+++Ω	REQUEST_ID	Number	☑	□	Request ID
Start Date	ΩStart_Date_△'+++Ω	START_DATE	Date	☑	☑	Start Date

Columns Hidden 3

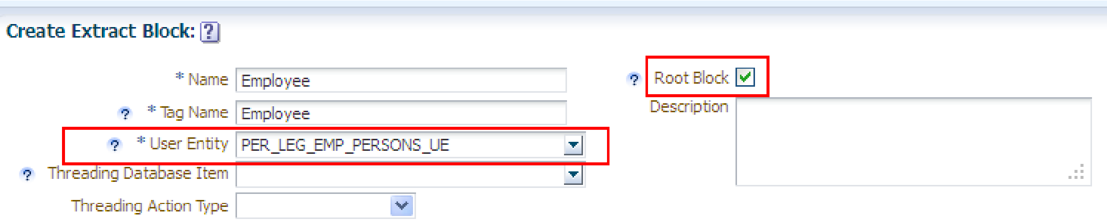
3 – Define Block

The basic Extract model uses a block to group the data retrieval. You need to define a block to fetch data and contexts for your records and elements. Blocks may be organized in a hierarchy or linked together using Block links. There will always be a root block which may contain multiple sub blocks.

In this example the root block will retrieve data at the employee level and therefore this block will use a User entity designed to retrieve employees. PER_LEG_EMP_PERSONS_UE

a) Click **Extract Block->Create** to create the root extract block

b) Select PER_LEG_EMP_PERSONS_UE as the User entity for this block, and make sure the Root Block is selected



Create Extract Block:

* Name Employee

* Tag Name Employee

* User Entity PER_LEG_EMP_PERSONS_UE

Threading Database Item

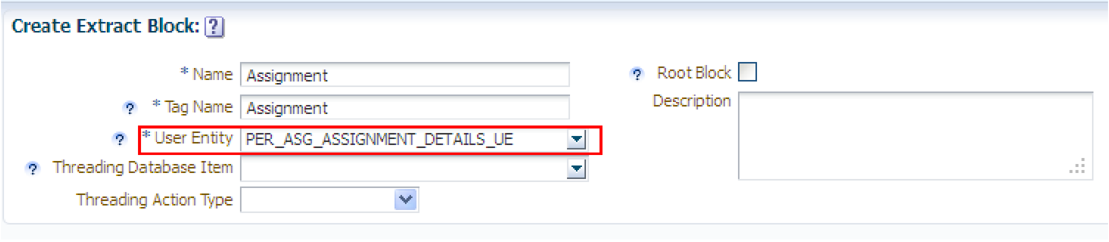
Threading Action Type

Root Block ☑

Description

The second level block will retrieve data at the assignment level, and therefore utilizes a user entity that retrieves assignment information using the root blocks employee level context.

c) Create second block for Assignment level data. Here choose a predefined User entity **PER_ASG_ASSIGNMENT_DETAILS_UE** to fetch Assignment data within the root blocks employee context.



Create Extract Block:

*Name Assignment

*Tag Name Assignment

*User Entity PER_ASG_ASSIGNMENT_DETAILS_UE

Threading Database Item

Threading Action Type

Root Block □

Description

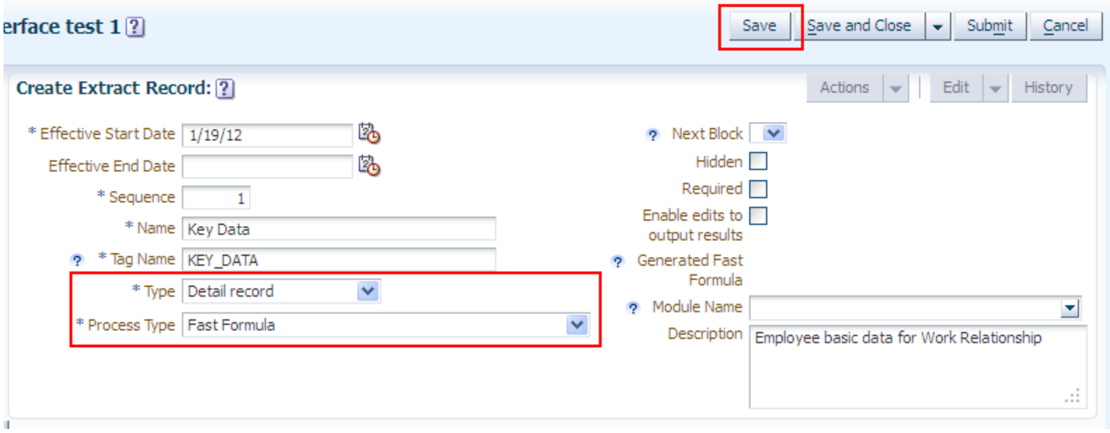
4 – Define Record

The record definition allows you to structure the output data appropriately. Records may be created at each level within the block structure. The blocks provide a container for grouping records.   The record provides a container for Extract elements, defining the record structure is important to avoid data redundancy.

In this example we will first create a record at the Employee level, and then a second record at the Assignment level.

a) Click **Employee** block in tree menu go to block detail page, use **Create** button to create new record

b) Choose **Detail record** for Type and **Fast Formula** for Process Type.



Create Extract Record:

*Effective Start Date 1/19/12

Effective End Date

*Sequence 1

*Name Key Data

*Tag Name KEY_DATA

*Type Detail record

*Process Type Fast Formula

Next Block

Hidden □

Required □

Enable edits to output results □

Generated Fast Formula

Module Name

Description Employee basic data for Work Relationship

c) Create record in **Assignment** block for Assignment level data

5 – Define Elements

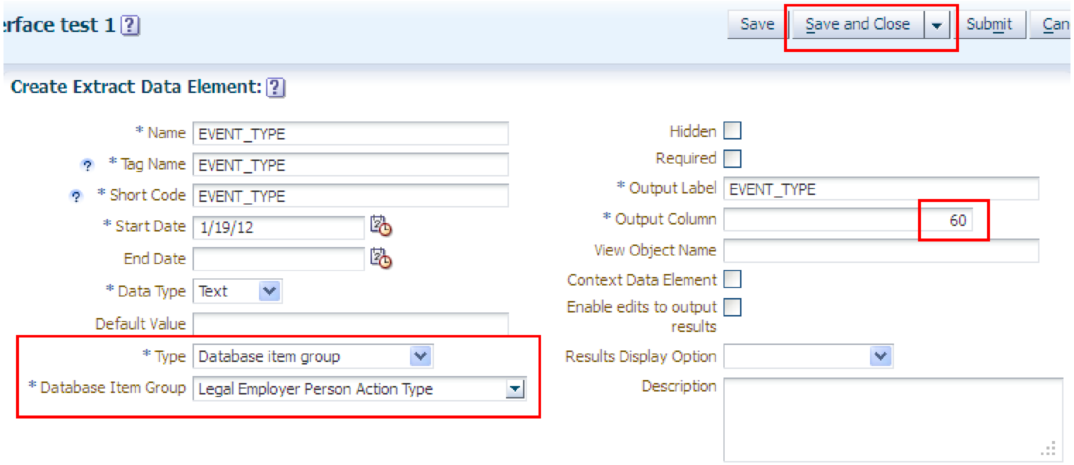
Within a record, elements may be defined to retrieve data. Data Base Item Groups (DBI Groups) are used to fetch data for an element. The Database Item Group must have the correct contexts for the record.

a) Click **Create** in Record detail information page.

b) Select Database item group for Type and select a suitable Database Item Group for current element.

c) Fill in all other required fields.

d) Click **Save and Close** to finish definition.



Create Extract Data Element:

* Name: EVENT_TYPE
* Tag Name: EVENT_TYPE
* Short Code: EVENT_TYPE
* Start Date: 1/19/12
End Date:
* Data Type: Text
Default Value:
* Type: Database item group
* Database Item Group: Legal Employer Person Action Type

Hidden □
Required □
* Output Label: EVENT_TYPE
* Output Column: 60
View Object Name:
Context Data Element □
Enable edits to output results □
Results Display Option:
Description:

6 – Configure block links

A block link may be used to link two blocks together matching the data using key data. To use this feature, two Blocks must be configured to use a DBI with matching key data to its user entity.

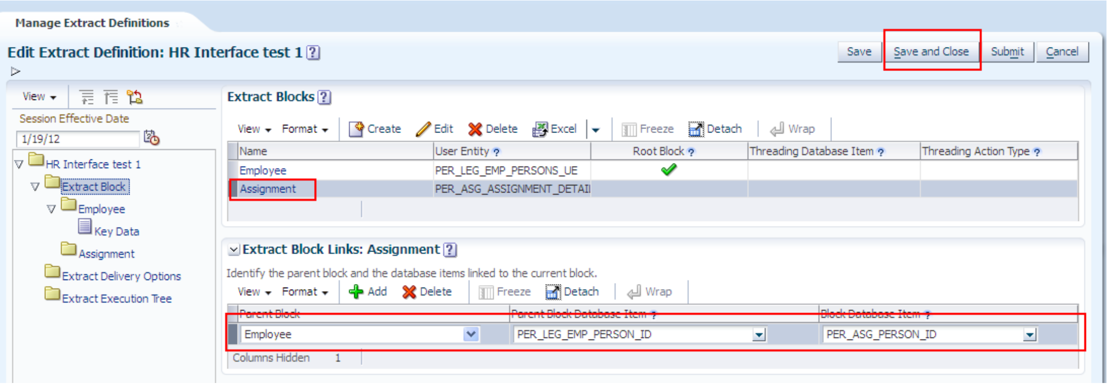
In this example we will use Person_ID to match data **between** the blocks we have previously defined. The User Entity for the root block has a DBI **Legal Employer Person Id**. Our sub block (At the assignment level) needs to choose User Entity which contains a DBI for **Person Id.**

a) Click **Extract Block** in tree menu, all blocks will then be displayed in the Right hand frame.

b) Select our **Assignment** level Block, you will notice that the **Extract Block Links: Assignment** will be displayed below block list.

c) We will now link the two blocks together, by choosing **Employee** for Parent Block, **PER_LEG_EMP_PERSON_ID** for Parent Block Database Item, **PER_ASG_PERSON_ID** for Block Database Item.

d) Click **Save and Close** to finish definition.



Edit Extract Definition: HR Interface test 1

Session Effective Date: 1/19/12

- HR Interface test 1
  - Extract Block
    - Employee
      - Key Data
    - Assignment
  - Extract Delivery Options
  - Extract Execution Tree

Extract Blocks

Name	User Entity	Root Block	Threading Database Item	Threading Action Type
Employee	PER_LEG_EMP_PERSONS_UE	✓		
Assignment	PER_ASG_ASSIGNMENT_DETAII			

Extract Block Links: Assignment

Identify the parent block and the database items linked to the current block.

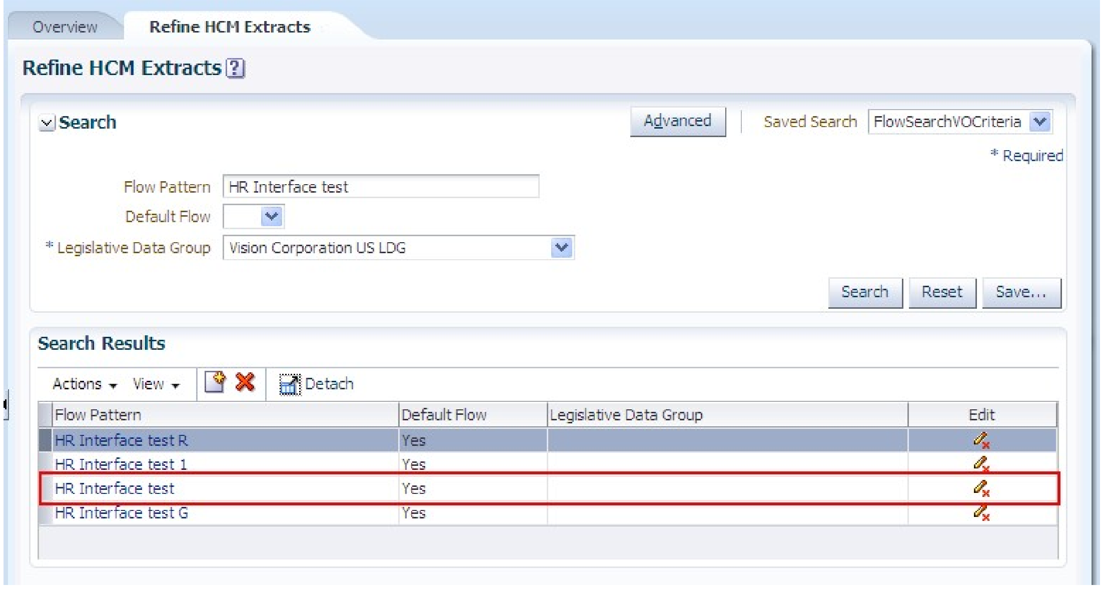
Parent Block	Parent Block Database Item	Block Database Item
Employee	PER_LEG_EMP_PERSON_ID	PER_ASG_PERSON_ID

Columns Hidden 1

Refining an Extract

After an extract has been defined the automatically created flow definition and parameters may be reviewed and revised through the use of the Refine Extracts UI.

1) After saving your extract you may navigate to the Refine ExtractUI, use the search to locate the correct flow pattern.



Saved Search: FlowSearchVOCriteria

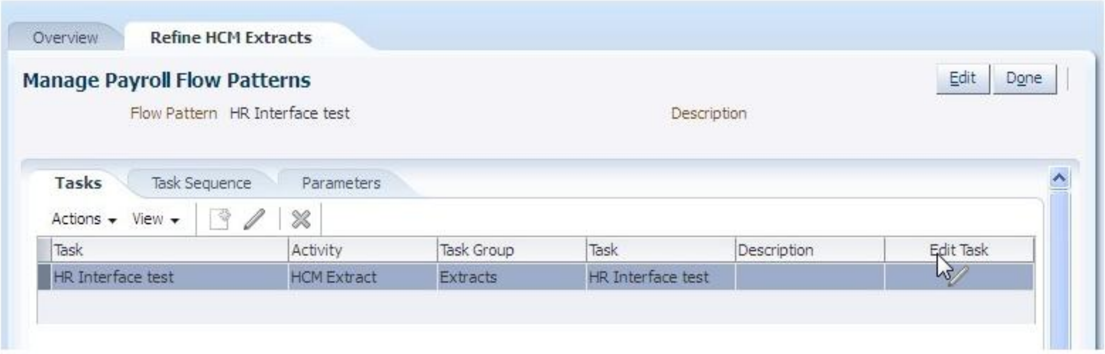
Flow Pattern: HR Interface test

Default Flow:

* Legislative Data Group: Vision Corporation US LDG

Flow Pattern	Default Flow	Legislative Data Group	Edit
HR Interface test R	Yes		
HR Interface test 1	Yes		
HR Interface test	Yes		
HR Interface test G	Yes		

2) You will then be presented with a screen that allows you to revise the Tasks and parameters for your extract

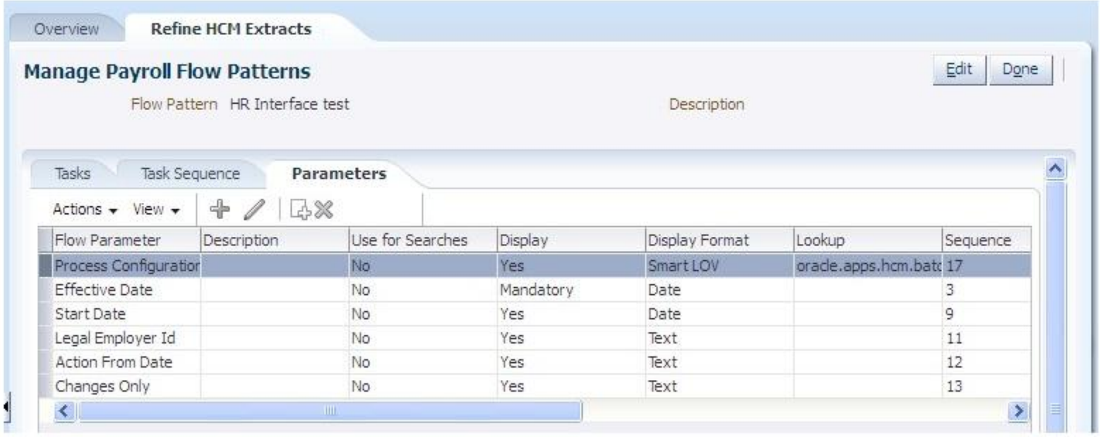


Flow Pattern: HR Interface test

Description:

Task	Activity	Task Group	Task	Description	Edit Task
HR Interface test	HCM Extract	Extracts	HR Interface test		

3) You may revise parameters from this screen



Manage Payroll Flow Patterns

Flow Pattern HR Interface test

Description

Flow Parameter	Description	Use for Searches	Display	Display Format	Lookup	Sequence
Process Configuratior		No	Yes	Smart LOV	oracle.apps.hcm.batc	17
Effective Date		No	Mandatory	Date		3
Start Date		No	Yes	Date		9
Legal Employer Id		No	Yes	Text		11
Action From Date		No	Yes	Text		12
Changes Only		No	Yes	Text		13

Running an Extract

This section will outline the steps required to execute a predefined extract. As part of the extract definition process a payroll flow and ESS submission task will be automatically created.

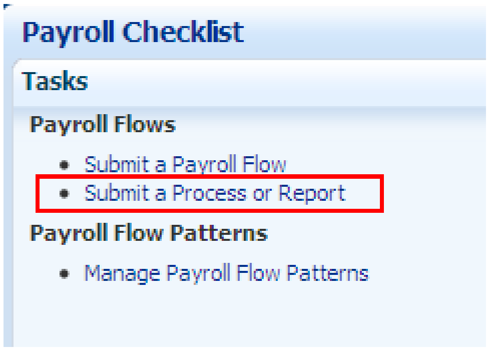
Note - There is no facility for updating an ESS/FLOW once the initial extract is created. If you need to add a parameter you will need to recreate your extract definition – this is a known issue and will be addressed in future release.

1 – Submit your extract

A block link may be used to link two blocks together matching the data using key data. To use this feature, two Blocks must be configured to use a DBI with matching key data to its user entity.

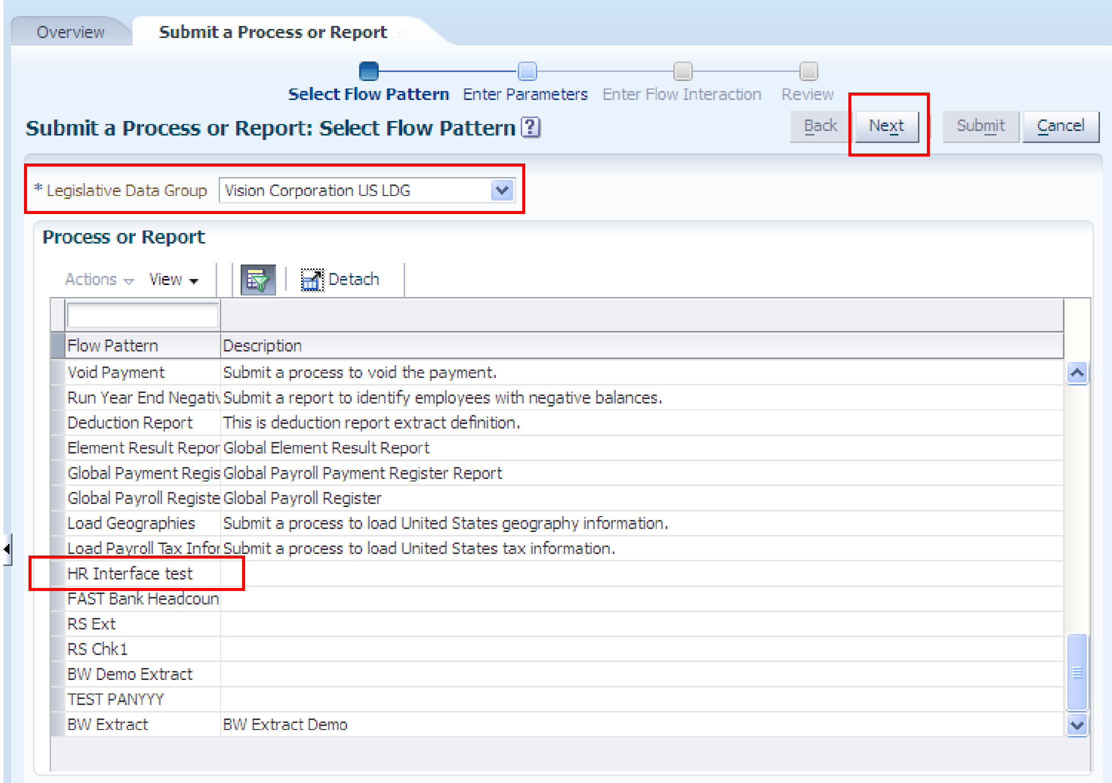
a) Login to **Hcm Payroll** server and select **Payroll Checklist** to Payroll Flow page.

b) Select **Tasks**-> **Payroll Flows**-> **Submit a Process or Report**



c) Choose **Legislative Data Group** in **Submit a Process or Report** page.

d) Select **HR Interface test** and click **Next**.



Submit a Process or Report: Select Flow Pattern

* Legislative Data Group: Vision Corporation US LDG

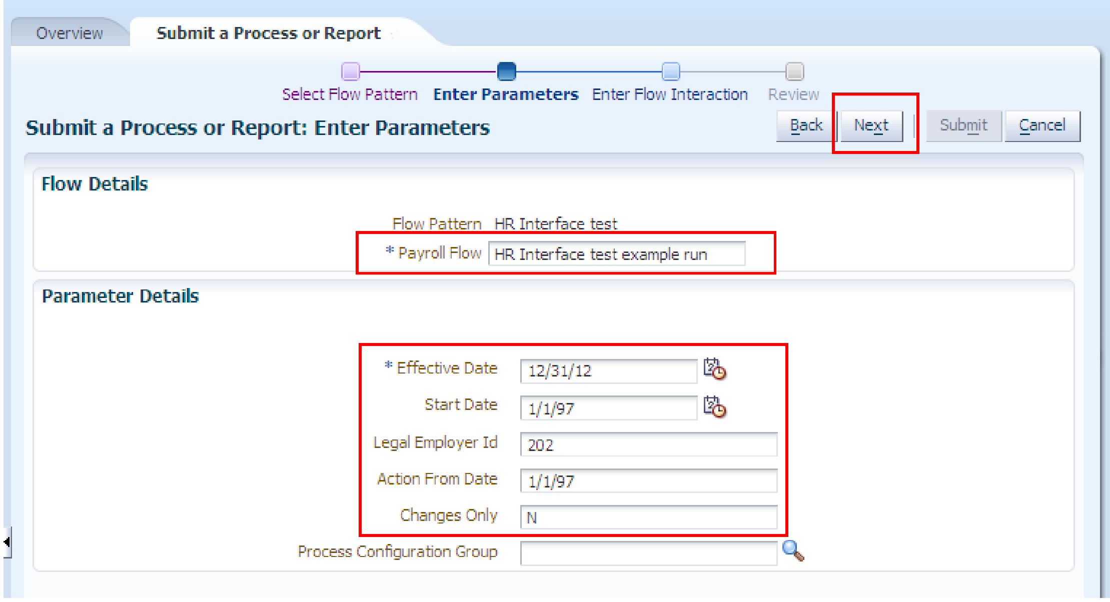
Process or Report

Flow Pattern	Description
Void Payment	Submit a process to void the payment.
Run Year End Negati	Submit a report to identify employees with negative balances.
Deduction Report	This is deduction report extract definition.
Element Result Repor	Global Element Result Report
Global Payment Regis	Global Payroll Payment Register Report
Global Payroll Registe	Global Payroll Register
Load Geographies	Submit a process to load United States geography information.
Load Payroll Tax Infor	Submit a process to load United States tax information.
HR Interface test	
FAST Bank Headcoun	
RS Ext	
RS Chk1	
BW Demo Extract	
TEST PANYYY	
BW Extract	BW Extract Demo

e) Fill **Payroll Flow** and other parameters.

f) Click **Next** to continue.

g) Click **Next** directly at **Enter Flow Interaction** step.



Submit a Process or Report: Enter Parameters

Flow Details

Flow Pattern HR Interface test

* Payroll Flow: HR Interface test example run

Parameter Details

* Effective Date: 12/31/12

Start Date: 1/1/97

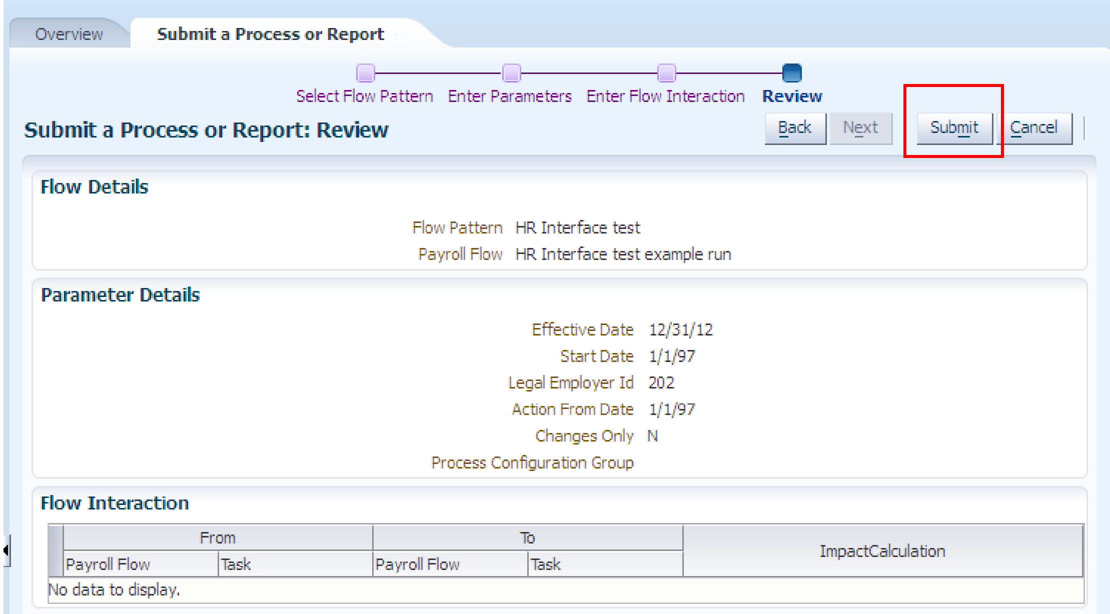
Legal Employer Id: 202

Action From Date: 1/1/97

Changes Only: N

Process Configuration Group:

h) Review parameters and click **Submit**.



Submit a Process or Report: Review

Flow Details

Flow Pattern	HR Interface test
Payroll Flow	HR Interface test example run

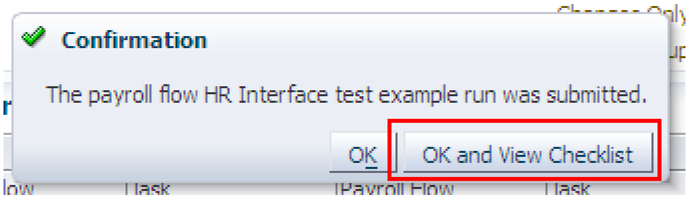
Parameter Details

Effective Date	12/31/12
Start Date	1/1/97
Legal Employer Id	202
Action From Date	1/1/97
Changes Only	N
Process Configuration Group	

Flow Interaction

From		To		ImpactCalculation
Payroll Flow	Task	Payroll Flow	Task	
No data to display.				

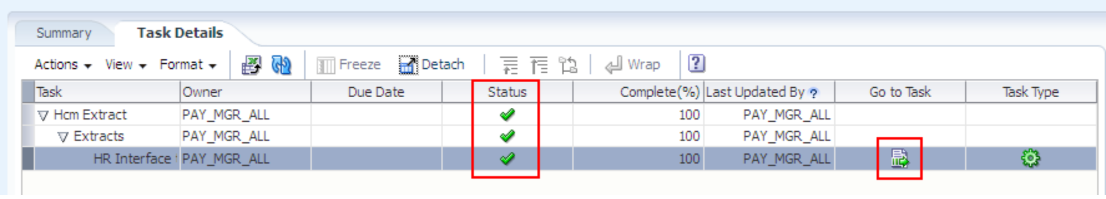
i) In the popup dialog, click **OK and View Checklist** go to running status page.



Confirmation

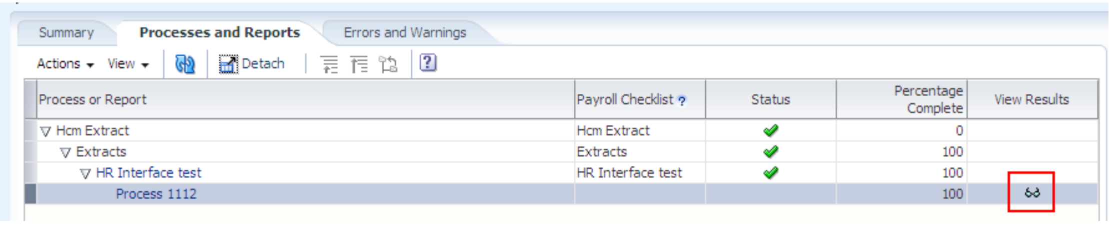
The payroll flow HR Interface test example run was submitted.

j) Make sure all Tasks are successful and click the icon in **Go to Task** column.



Task	Owner	Due Date	Status	Complete(%)	Last Updated By	Go to Task	Task Type
▽ Hcm Extract	PAY_MGR_ALL		✓	100	PAY_MGR_ALL		
▽ Extracts	PAY_MGR_ALL		✓	100	PAY_MGR_ALL		
HR Interface	PAY_MGR_ALL		✓	100	PAY_MGR_ALL		⚙

k) Click icon in **View Result** column to get running result.



Process or Report	Payroll Checklist	Status	Percentage Complete	View Results
▽ Hcm Extract	Hcm Extract	✓	0	
▽ Extracts	Extracts	✓	100	
▽ HR Interface test	HR Interface test	✓	100	
Process 1112			100	👓

Incremental Extracts

An incremental extract identifies data that has changed between extracts at the root block level. If a difference is found between runs then the entire data for that person will be included in the extract.

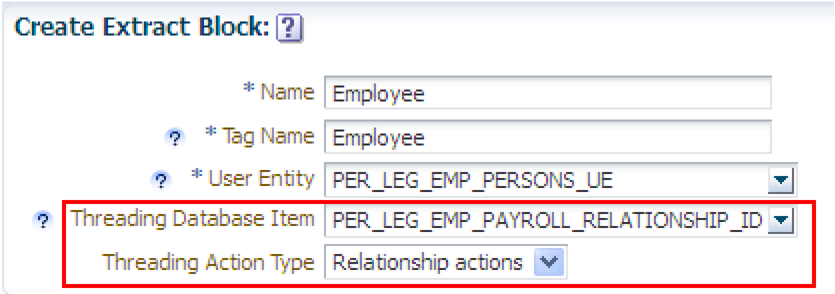
To use this feature, you simply need to define an Extract Parameter **Changes Only** and set it as **Y** to run Extract.

This feature will only do the comparison for data extracted at the Payroll Relationship level or below.

Incremental extracts will only take effect on multi-thread block, root block should have following settings:

a) Threading Action Type:**Relationship Actions**

b) Threading Database Item: DBI name for **PAY_PAY_RELATIONSHIPS_DN.PAYROLL_RELATIONSHIP_ID**



Create Extract Block:

* Name	Employee
* Tag Name	Employee
* User Entity	PER_LEG_EMP_PERSONS_UE
Threading Database Item	PER_LEG_EMP_PAYROLL_RELATIONSHIP_ID
Threading Action Type	Relationship actions

Oracle Corporation
World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065
U.S.A.

Worldwide Inquiries:
Phone: +1.650.506.7000
Fax: +1.650.506.7200

oracle.com