Advanced Reporting

Guide to the manual method for creating Change Reports

Manual method for Change Reporting

This document aims to guide a user through creating a “manual” Change Report (ie: without using the Change Reporting function made available in the new UI released in May 2015 (b1505)).

The current Change Reporting function allows for reporting changes on a single column only. Given this, the main benefit for manually creating these reports (over using the Change Reporting functionality) is to create a Change Report that shows changes in more than one column (which this guide will cover).

New vs Classic Advanced Reporting User Interfaces

This guide assumes a basic level of knowledge in Advanced Reporting and contains two sections, each one describing the manual Change Reporting method for each UI of Advanced Reporting:

- The ‘new’ Advanced Reporting UI was made available in the May 2015 release and most instances implemented since then will be on this UI, in addition to any existing instance that has requested to be moved to this new UI

- Most instances implemented prior to the May 2015 release will be using the ‘classic’ Advanced Reporting UI

If you are unsure which UI your instance is using, please refer to the screen shots in each section and choose the section which contains screen shots that match your instance.

This guide splits out how to duplicate the Change Reporting table in each of the UI’s (new and classic) – the last section (applying necessary filters) applies to both UI’s.

Initial Notes

The manual change reporting method works from the premise of a “base” or anchor table and then joining two instances of the table containing the change reporting columns onto this base table. Two separate instances of a table allows the query to apply a different date filter to each instance, which in turn retrieves the current and previous values.

Other tables unrelated to the change column can be pulled into the query for additional report data if required.

The following slides will refer to “current” and “previous” values – however the same method can apply for historical (rather than the current) values simply by adjusting the date filters on the relevant tables.

This method shows changes that occurred within a “day” timeframe. Should an employee have more than one change in a single day, only the last record in that day will be captured in the results.

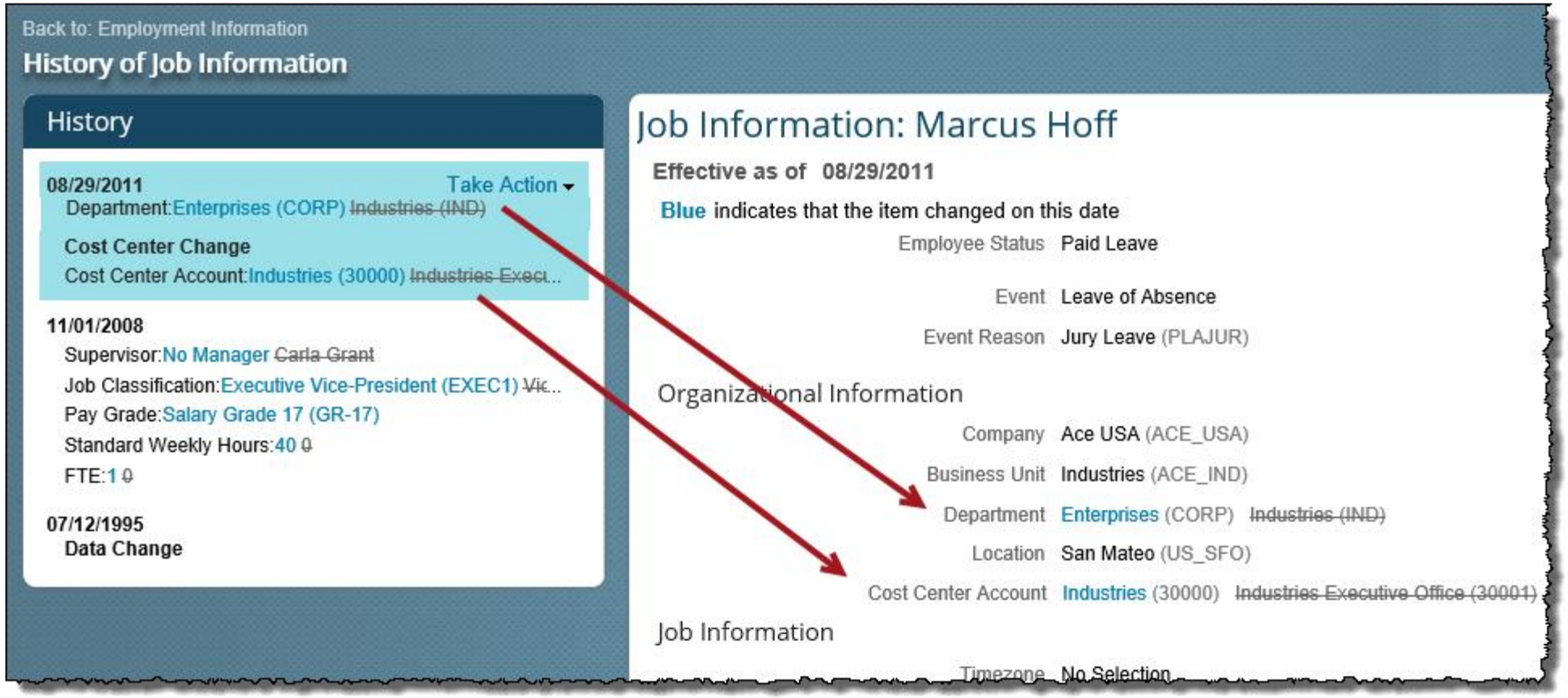
Duplicating the Change Reporting table in the ‘new’ Advanced Reporting UI

*Please skip to the **Duplicating the Change Reporting table in the ‘classic’ UI** section if you are not using the b1505 UI.*

EC Job Information Changes

New UI

For this example we will look at the employee Marcus Hoff. He is currently in the *Enterprises* Department and the *Industries* Cost Center, having moved from *Industries* and *Industries Executive Office* respectively.



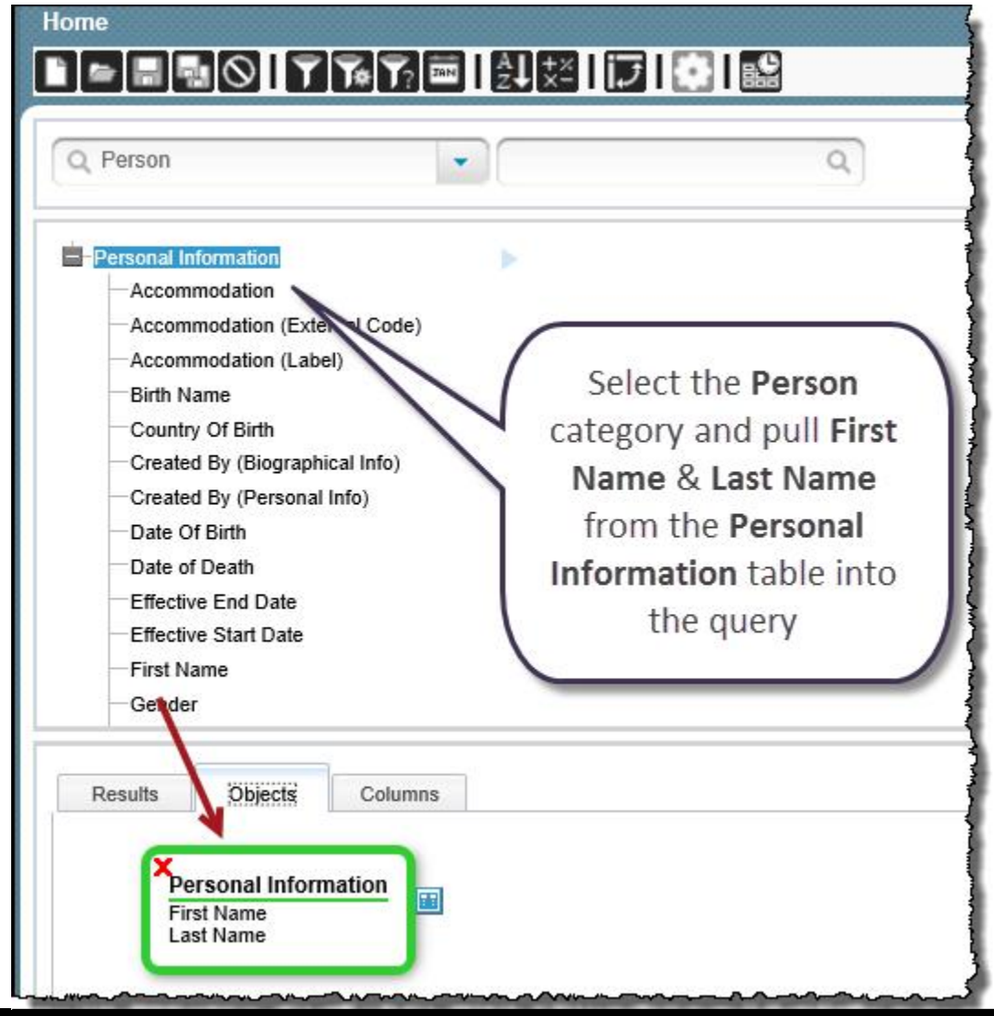
Begin the query

New UI

To start, we will pull in data from the table which will be the ‘anchor’ table.

Typically this will be the Person > Personal Information table for queries that are Change Reporting on employee Job data, or reporting on employee Compensation data.

Select the **Person** category from the drop down and open the **Personal Information** table. Pull in the columns required from this table.

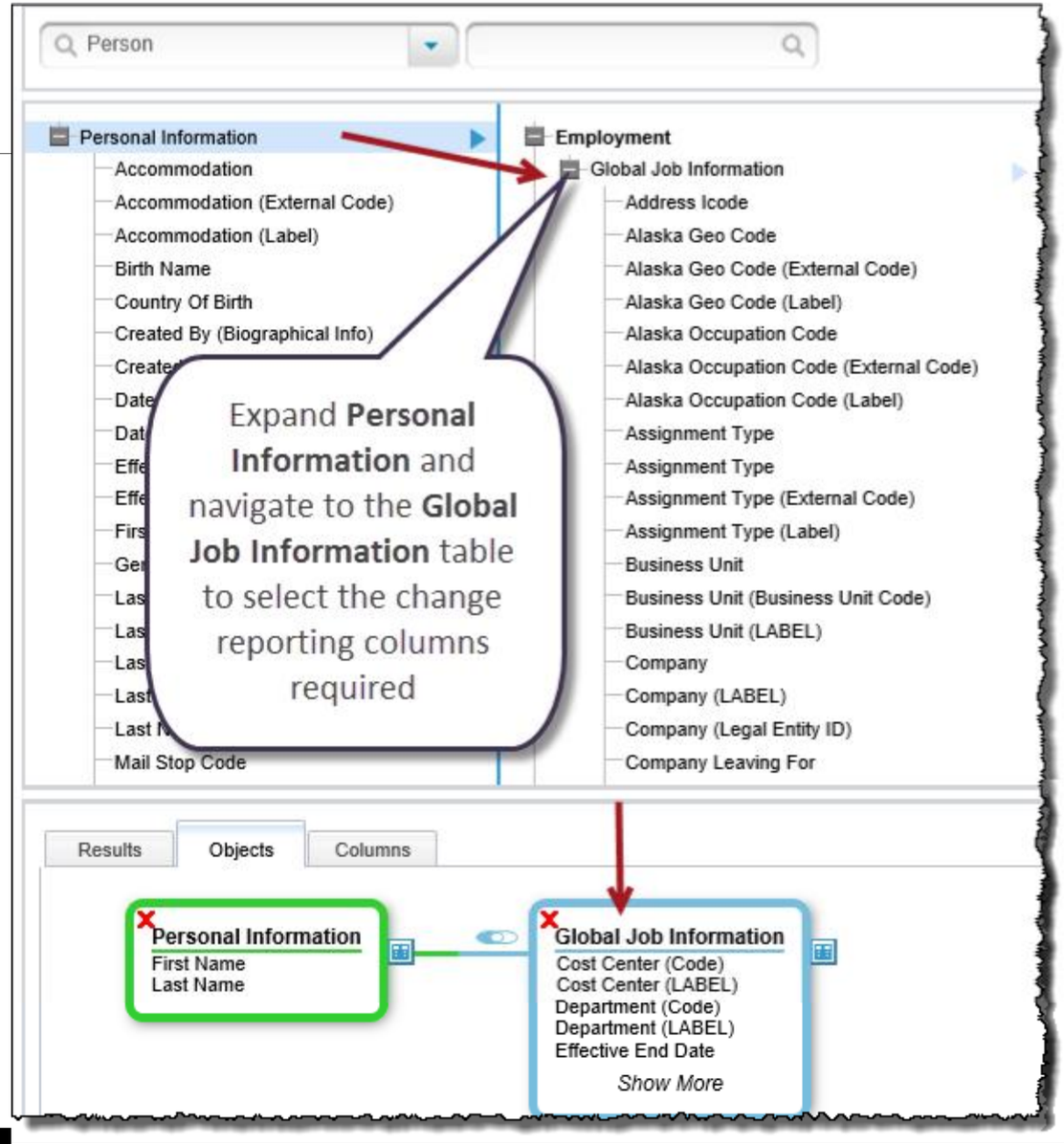


Pull in the first “change” table

New UI

Navigate to the **Global Job Information** table from the **Personal Information** table and pull in the columns required for change reporting.

In this example we are looking at changes on the **Department** and **Cost Center** columns. The **Effective Start/End Dates** have also been selected so that we can see the “timeline” in the finished report.

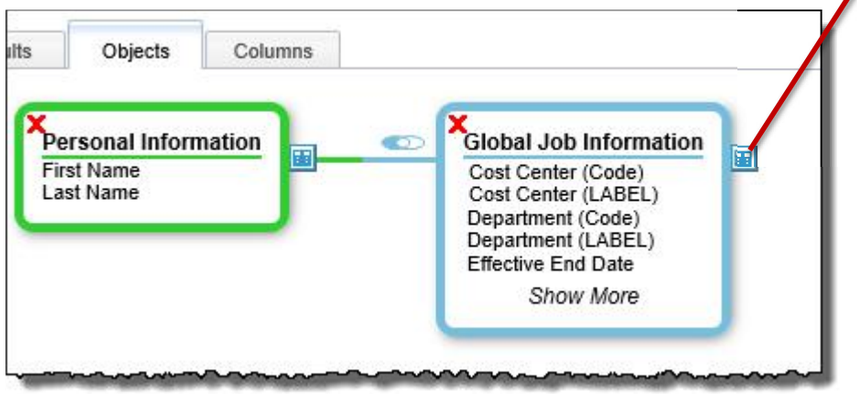


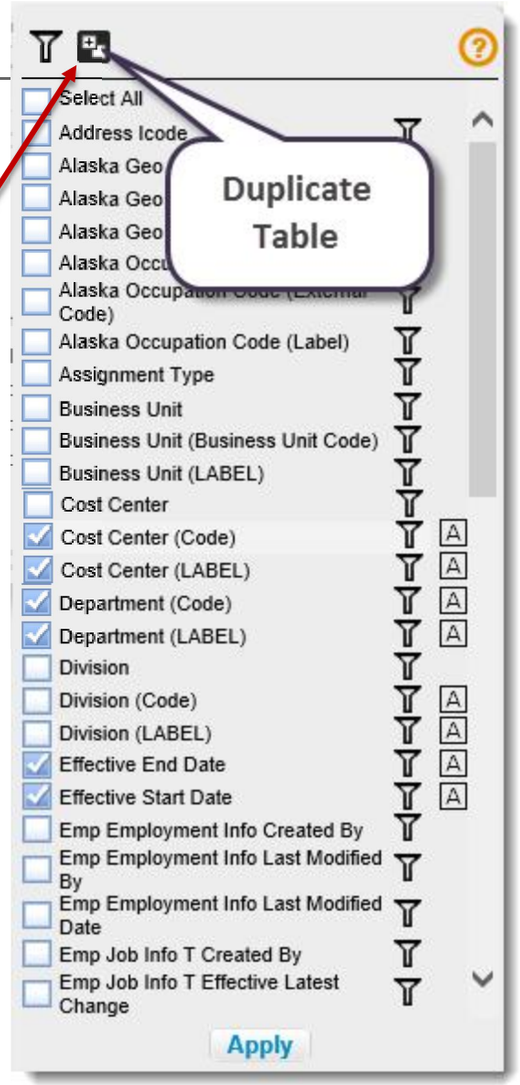
Duplicate the first “change” table

New UI

We now need to create a second instance of the table containing the “change” columns.

Open the column menu for the **Global Job Information** table and click **Duplicate Table**.



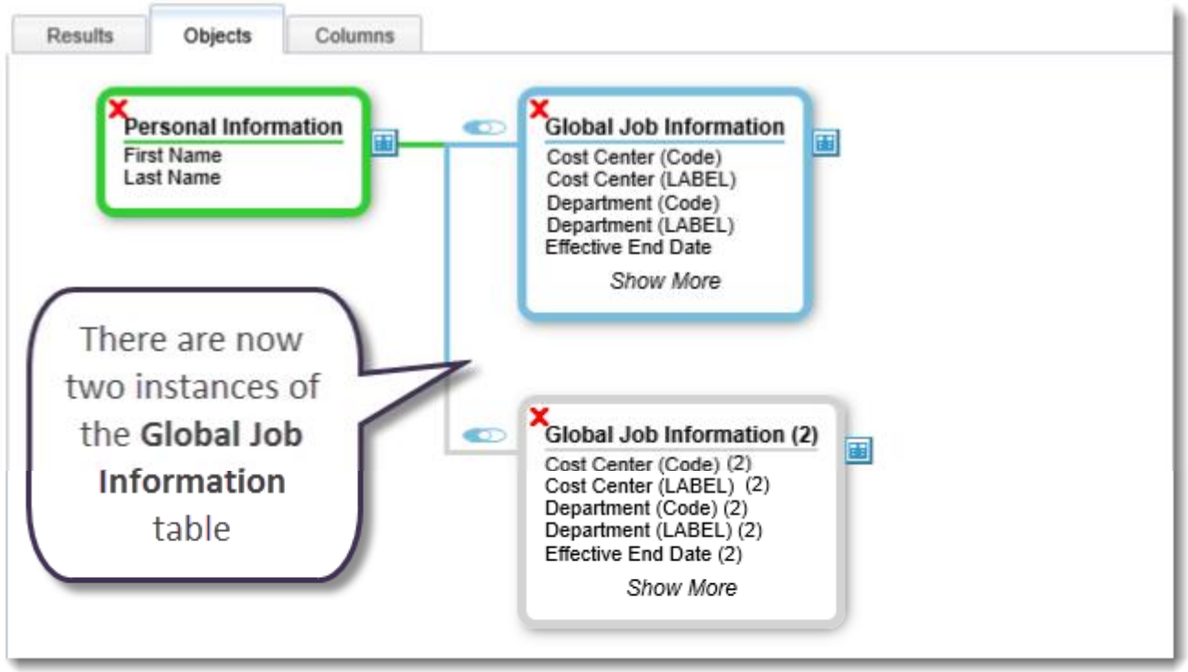


Second instance of the “change” table is now created

New UI

The **Global Job Information** table has been duplicated, with the identical column settings as the original.

Global Job Information will be the table retrieving the “current” Job information and **Global Job Information (2)** will be the table retrieving the “previous” Job information. So that we can more easily identify which **Department** + **Cost Center** is the current and which **Department** + **Cost Center** is the previous, we will rename the columns in each table appropriately.

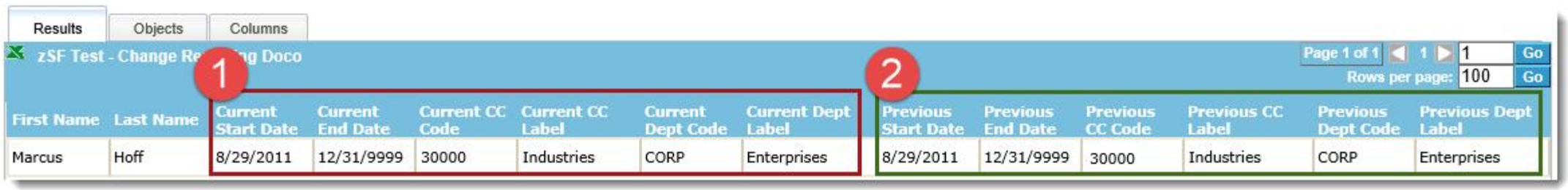


Checkpoint

New UI

We now have a report output that is duplicating the current Department and current Division for each employee. This occurs because both instances of the **Global Job Information** table defaults to the "current date". As you can see in the first record, the columns in Group 1 have identical results as the columns in Group 2.

Please jump to the **Finalize the manual Change Reporting query by adding the required filters** section where we will apply date filters to retrieve the proper current & previous results for Department and Cost Center.



First Name	Last Name	Current Start Date	Current End Date	Current CC Code	Current CC Label	Current Dept Code	Current Dept Label	Previous Start Date	Previous End Date	Previous CC Code	Previous CC Label	Previous Dept Code	Previous Dept Label
Marcus	Hoff	8/29/2011	12/31/9999	30000	Industries	CORP	Enterprises	8/29/2011	12/31/9999	30000	Industries	CORP	Enterprises

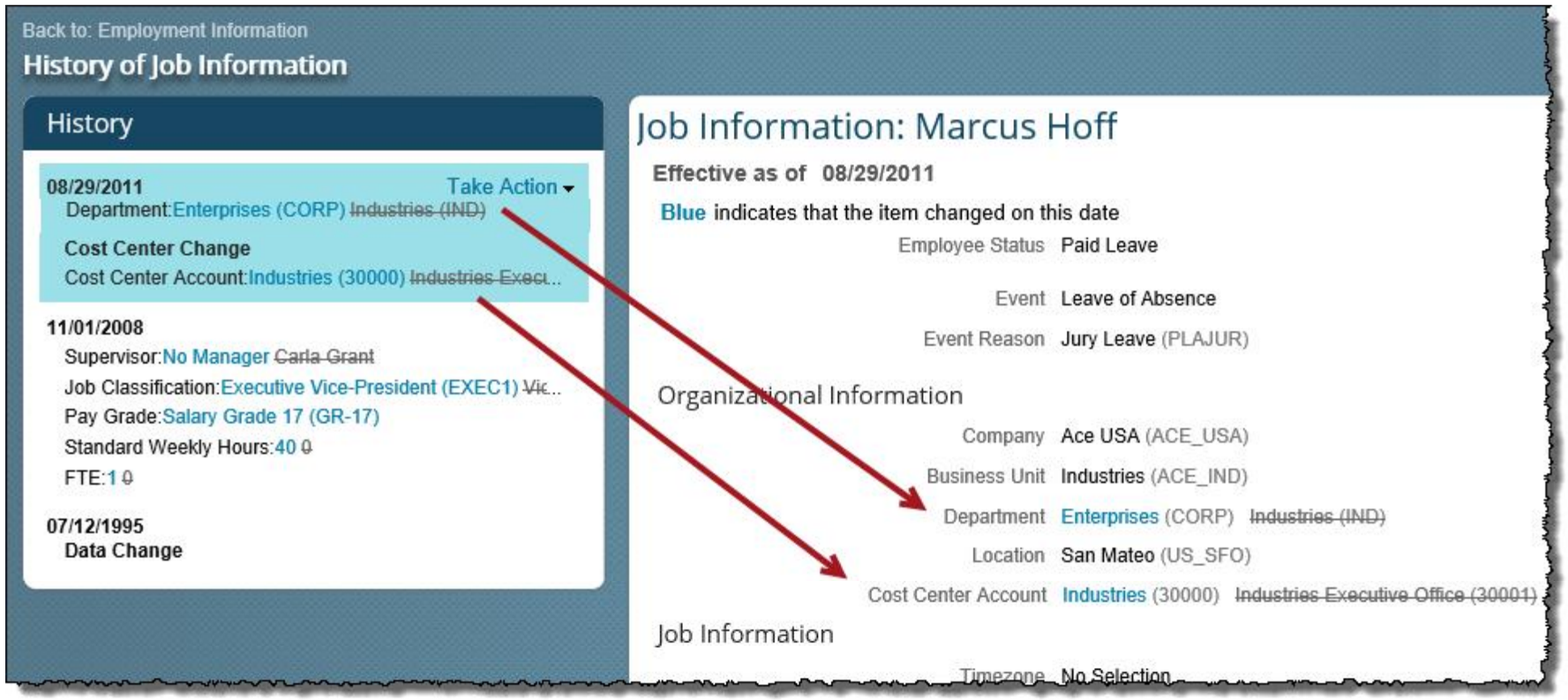
Duplicating the Change Reporting table in the ‘classic’ Advanced Reporting UI

*Please jump back to the **Duplicating the Change Reporting table in the ‘new’ UI** section if you are using the b1505 UI.*

EC Job Information Changes

Classic UI

For this example we will look at the employee Marcus Hoff. He is currently in the *Enterprises* Department and the *Industries* Cost Center, having moved from *Industries* and *Industries Executive Office* respectively.

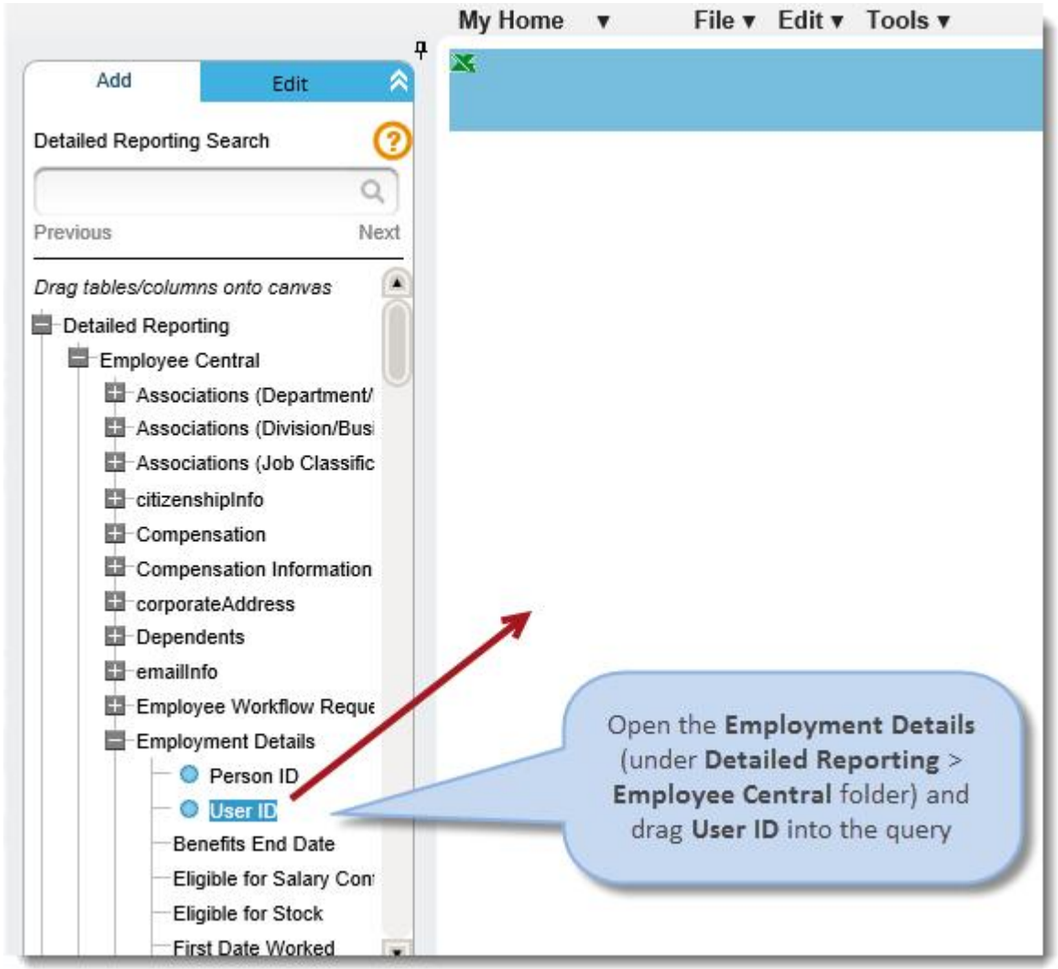


Begin the query

Classic UI

To start, we will pull in data from the table which will be the ‘anchor’ table. Typically this will be the **Employment Details** table for queries that are Change Reporting on employee Job data, or reporting on employee Compensation data.

Open the **Employment Details** table under **Detailed Reporting** > **Employee Central** and drag on **User ID** (and any other required columns from this table).



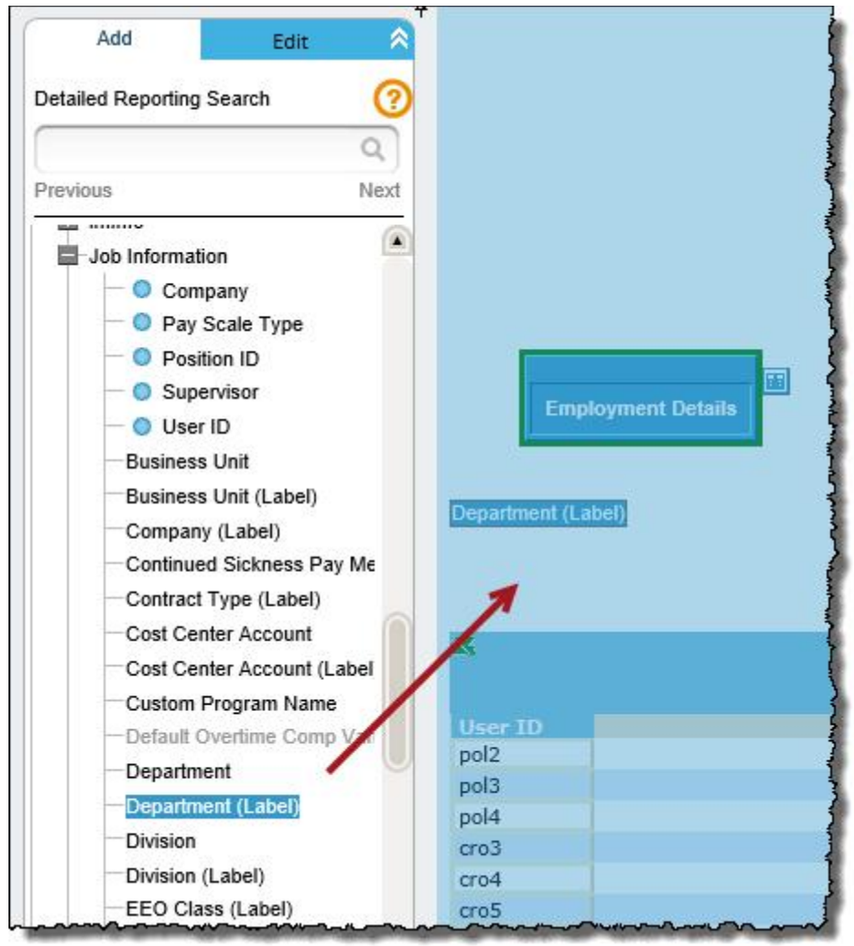
Pull in the first “change” table

Classic UI

Navigate to the **Job Information** table (also under **Detailed Reporting** > **Employee Central**).

In this example we are looking at changes on the **Department** and **Cost Center** columns.

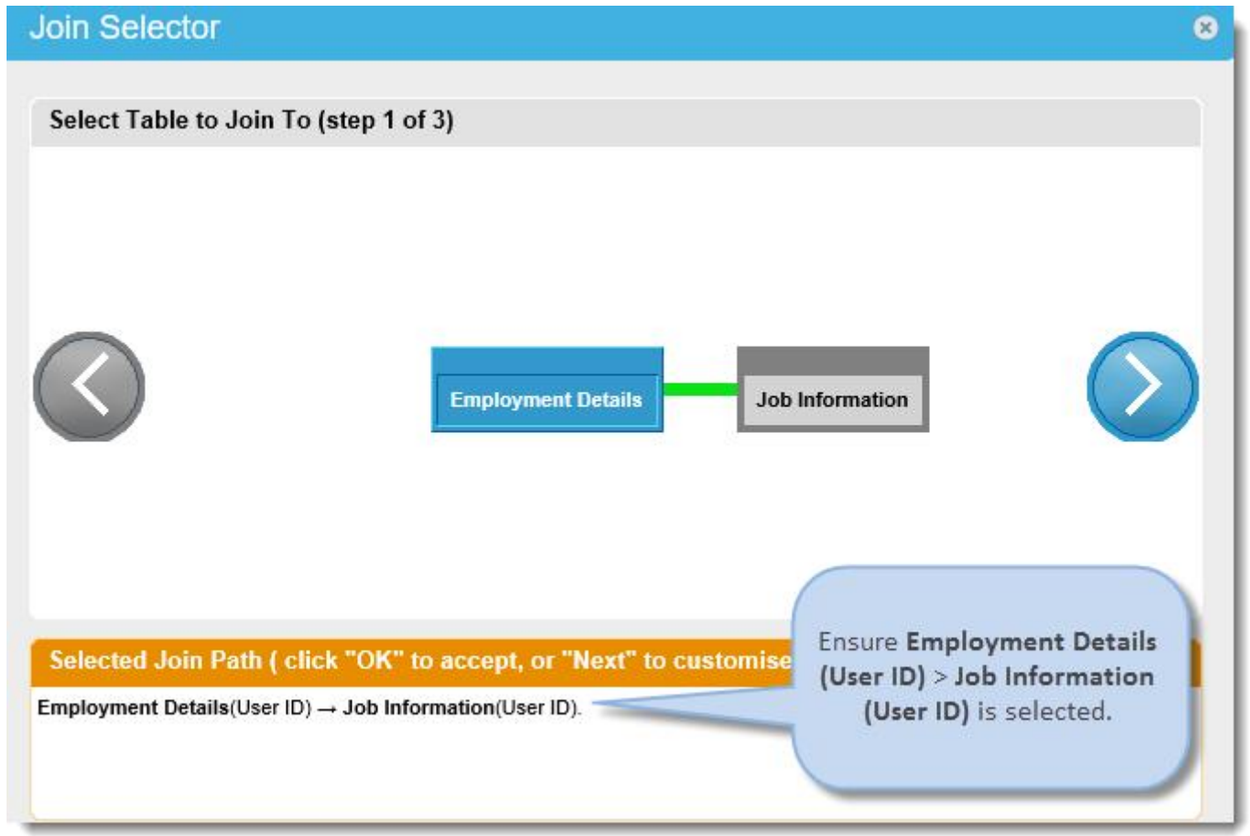
Ensure the **Advanced** view is enabled and pull the **Department (Label)** column onto the canvas to add the **Job Information** table into the query.



Configure the join for the first “change” table

Classic UI

As we are in **Advanced** view, the **Join Selector** dialog will open **–** ensure the User ID > User ID join is selected and click **OK**.

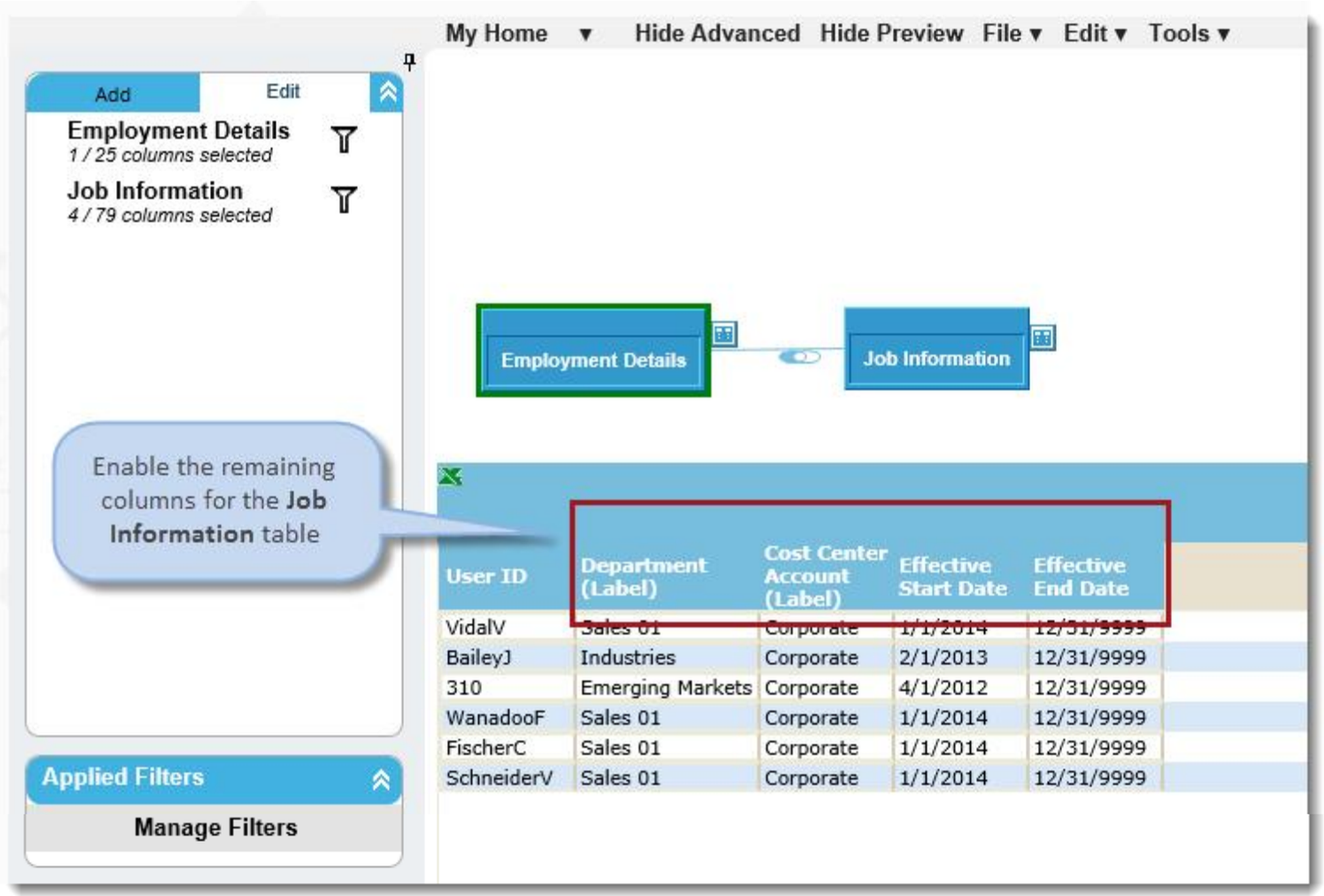


Configure the columns for the first “change” table

Classic UI

Enable the remaining required columns for the **Job Information** table:

- Effective Start Date
- Effective End Date
- Cost Center (Label)

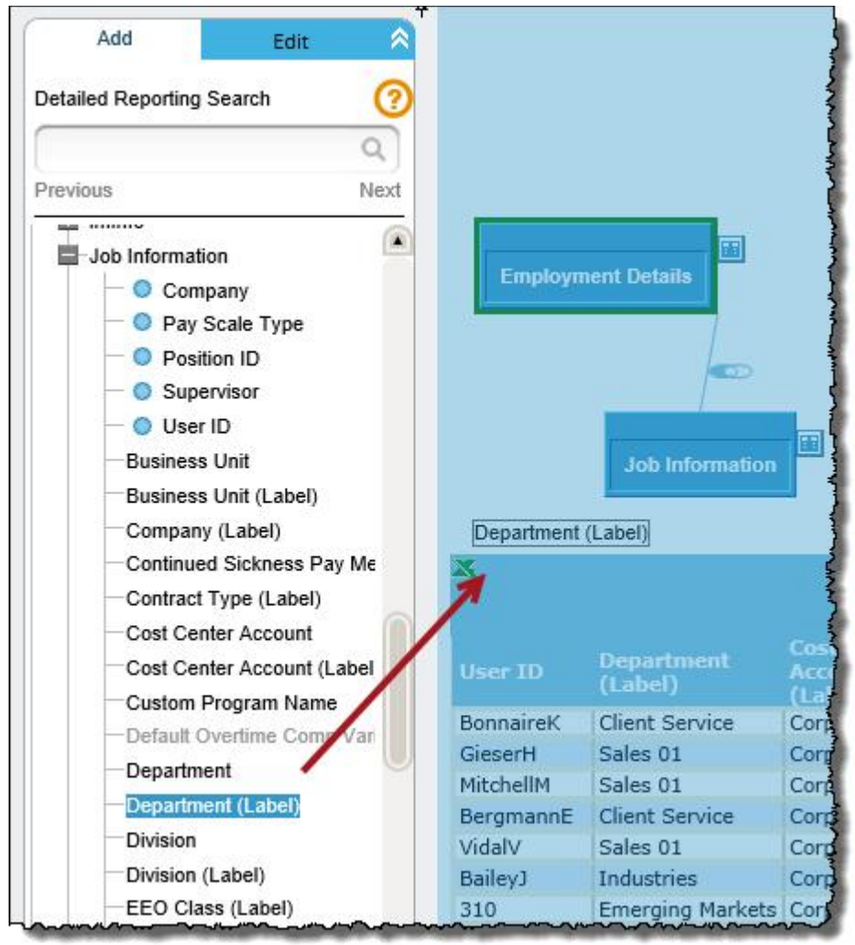


Duplicate the first “change” table

Classic UI

We now repeat the process of pulling in the **Job Information** table to create a second instance of the table containing the “change” columns.

Return to the column menu and drag the **Department (Label)** column again onto the canvas.

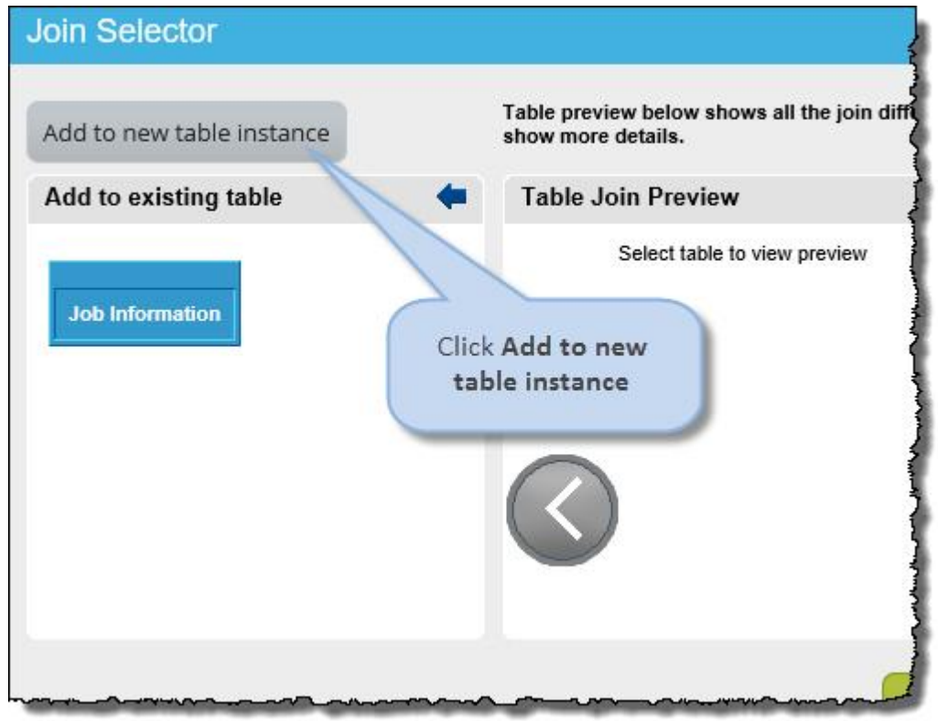


Make a new instance (duplicate) of the “change” table

Classic UI

When columns from a table that already exists in the query are pulled into the query, you will be offered the choice to add those columns to the existing table – or to create a new instance of the table.

For Change Reporting purposes we need to create a new instance of the **Job Information** table, so in the **Join Selector** dialog click **Add to new table instance**.

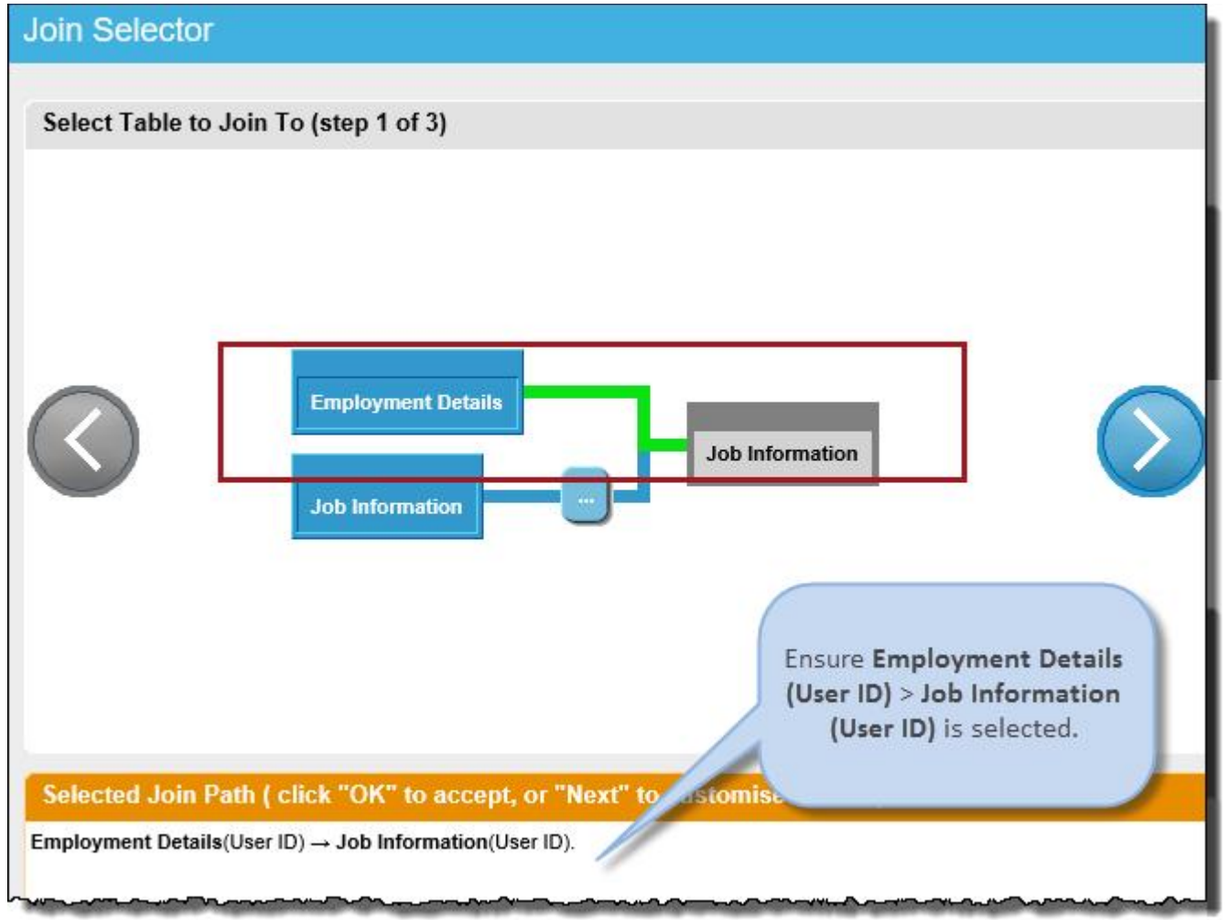


Configure the join for the duplicate “change” table

Classic UI

Ensure the same join is configured as for the first change table. Here we will again ensure the User ID > User ID join is selected and click **OK**.

Note the table layout graphic is slightly different here, showing the first **Job Information** table that we have already pulled into the query.

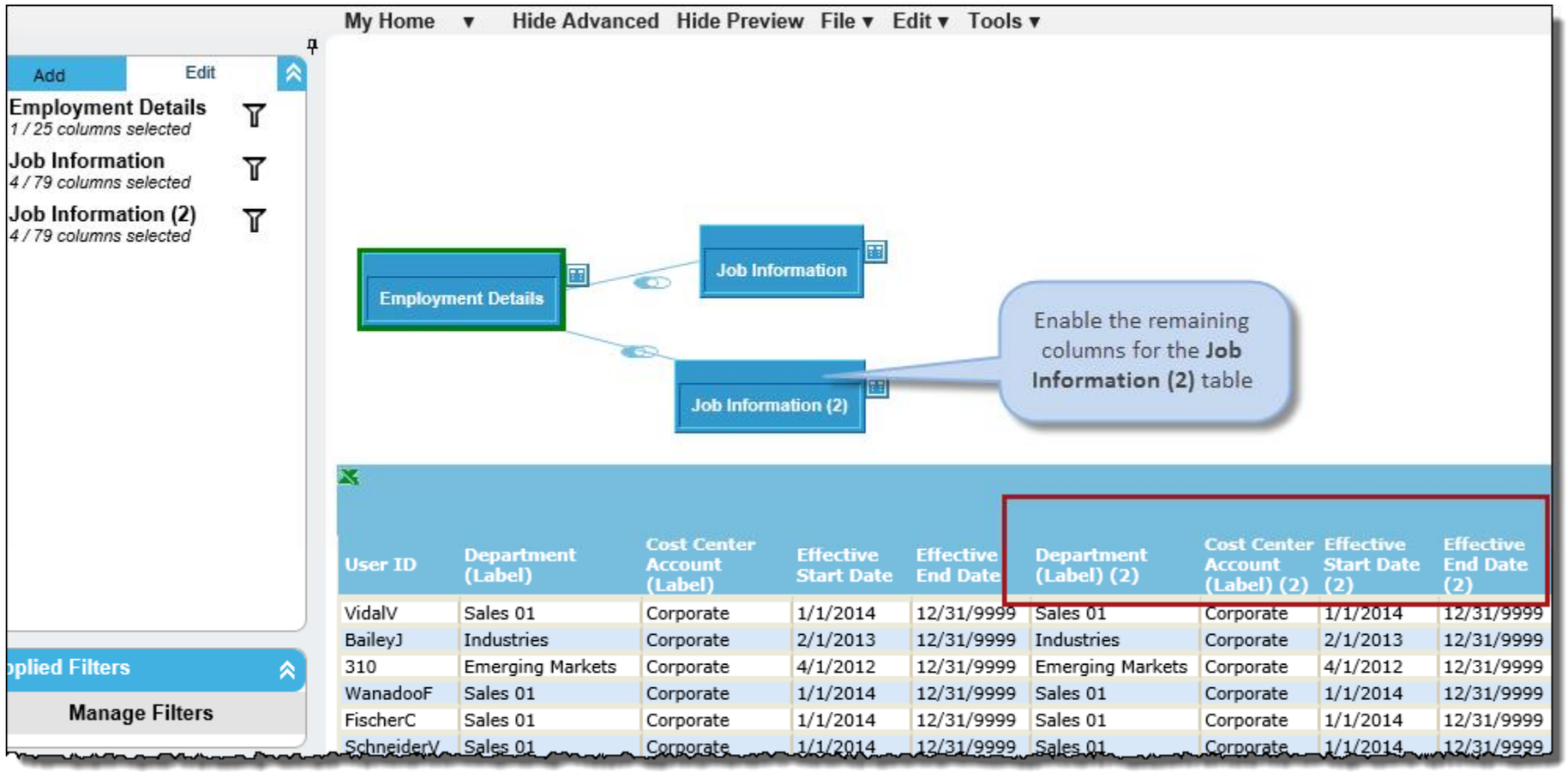


Second instance of the “change” table is now created

Classic UI

Another instance of the **Job Information** table is now available in the query, modify it’s columns to switch on:

- Effective Start Date
- Effective End Date
- Cost Center (Label)

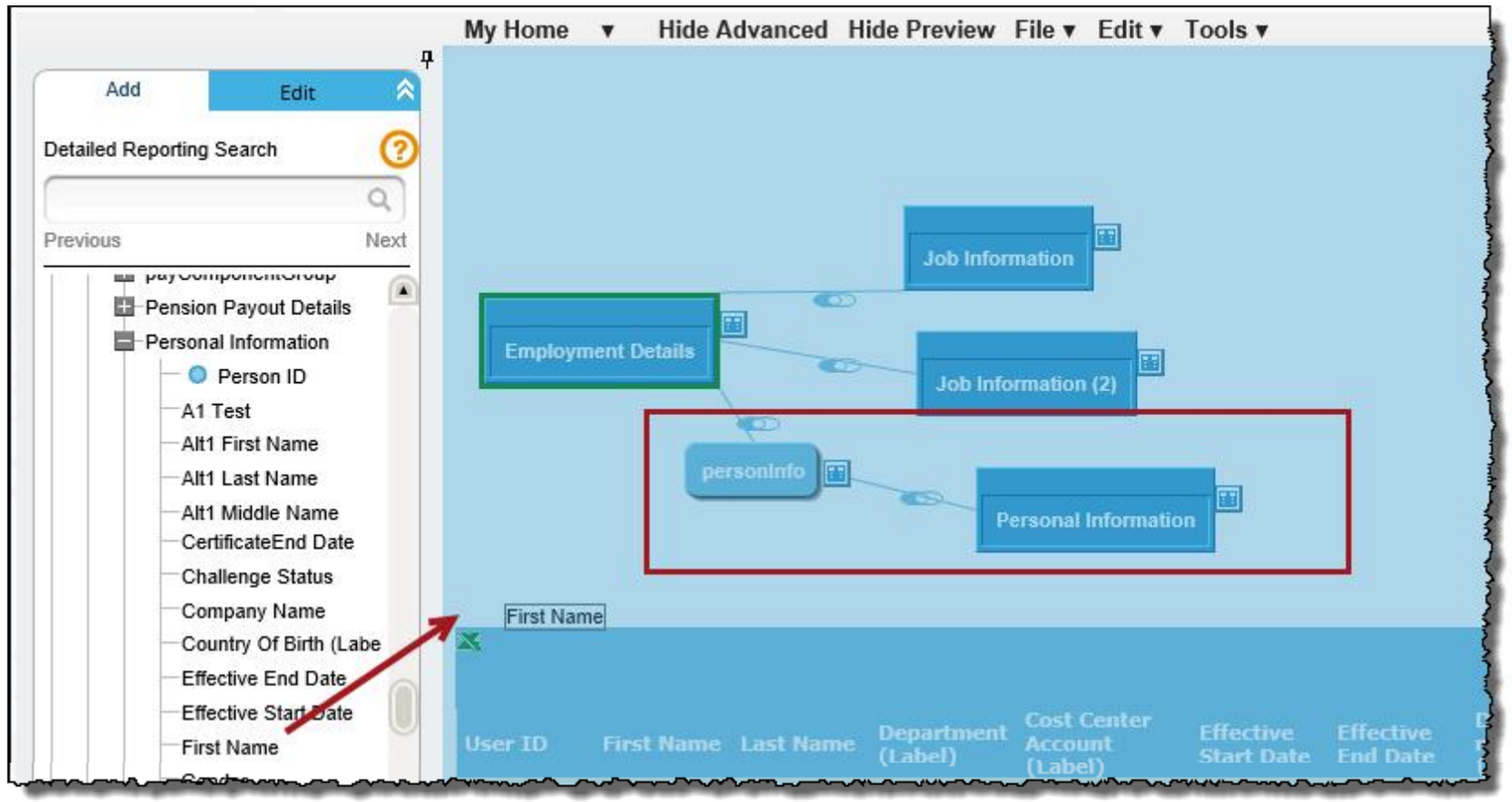


Pull in other tables as required

Classic UI

We would also like the **First** and **Last Name** in this query.

Pull these in from the **Personal Information** table, joining to the **Employment Details** table via the **Person** (or **Biographical**) **Info** table on the **Person ID**.



Checkpoint

Classic UI

We now have a report output that is duplicating the current Department and current Division for each employee. This occurs because both instances of the **Job Information** table defaults to the "current date". As you can see in the first record, the columns in Group 1 have identical results as the columns in Group 2.

In the next section we will apply date filters to retrieve the proper current & previous results for Department and Cost Center.