Lookup Transformation is Passive/Active and it can be Connected/Unconnected.

A Lookup transformation can used to look up data in a flat file, relational table, view, or synonym. You can import a lookup definition from any flat file or relational database to which both the PowerCenter Client and Integration Service can connect. You can also create a lookup definition from a source qualifier. You can use multiple Lookup transformations in a mapping.

The Integration Service queries the lookup source based on the lookup ports in the transformation and a lookup condition. The Lookup transformation returns the result of the lookup to the target or another transformation. You can configure the Lookup transformation to return a single row or multiple rows.

Perform the following tasks with a Lookup transformation:

* **Get a related value.** Retrieve a value from the lookup table based on a value in the source. For example, the source EMP table having DEPTNO but we don't have DNAME. We can get it by doing lookup on DEPT table based on condition DEPT.DEPTNO=EMP.DEPTNO.
* **Get multiple values**. Retrieve multiple rows from a lookup table. For example, the DEPT table having DEPTNO but we don't have EMPNO's in that department, we can return all employees in a department.
* **Perform a calculation.** Retrieve a Employees who's SAL > 2000 from a lookup table. We will be using Lookup SQL Override query.
* **Update slowly changing dimension tables.** This is used to determine whether rows exist in a target.

**Configure the Lookup transformation to perform the following types of lookups:**

* Relational or flat file lookup. Perform a lookup on a flat file or a relational table. When you create a Lookup transformation using a relational table as the lookup source, you can connect to the lookup source using ODBC and import the table definition as the structure for the Lookup transformation. When you create a Lookup transformation using a flat file as a lookup source, the Designer invokes the Flat File Wizard.
* Pipeline lookup. Perform a lookup on application sources such as a JMS, MSMQ, or SAP. Drag the source into the mapping and associate the Lookup transformation with the source qualifier. Configure partitions to improve performance when the Integration Service retrieves source data for the lookup cache.
* Connected or unconnected lookup. A connected Lookup transformation receives source data, performs a lookup, and returns data to the pipeline. An unconnected Lookup transformation is not connected to a source or target. A transformation in the pipeline calls the Lookup transformation with a :LKP expression. The unconnected Lookup transformation returns one column to the calling transformation.
* Cached or uncached lookup. Cache the lookup source to improve performance. If you cache the lookup source, you can use a dynamic or static cache. By default, the lookup cache remains static and does not change during the session. With a dynamic cache, the Integration Service inserts or updates rows in the cache. When you cache the target table as the lookup source, you can look up values in the cache to determine if the values exist in the target. The Lookup transformation marks rows to insert or update the target.

**Lookup Source Types**

When we create a Lookup transformation, we can choose a relational table, flat file, or a source qualifier as the lookup source.

**Relational Lookups**

When we create a Lookup transformation using a relational table as a lookup source, we can connect to the lookup source using ODBC and import the table definition as the structure for the Lookup transformation.

Use the following options with relational lookups:

* Override the default SQL statement to add a WHERE clause or to query multiple tables.
* Sort null data high or low, based on database support.
* Perform case-sensitive comparisons based on the database support.

**Flat File Lookups**

When we create a Lookup transformation using a flat file as a lookup source, select a flat file definition in the repository or import the source when you create the transformation. When you import a flat file lookup source, the Designer invokes the Flat File Wizard.

Use the following options with flat file lookups:

* Use indirect files as lookup sources by configuring a file list as the lookup file name.
* Use sorted input for the lookup.
* Sort null data high or low.
* Use case-sensitive string comparison with flat file lookups.

**Using Sorted Input**

When we configure a flat file Lookup transformation for sorted input, the condition columns must be grouped. If the condition columns are not grouped, the Lookup transformation returns incorrect results. For optimal caching performance, sort the condition columns.

For example, a Lookup transformation has the following condition:

OrderID = OrderID1 CustID = CustID1

In the flat file lookup source, the keys are grouped, but not sorted. The Integration Service can cache the data, but performance may not be optimal.

If we choose sorted input for indirect files, the range of data must not overlap in the files.

**Connected and Unconnected Lookups**

We can configure a connected Lookup transformation to receive input directly from the mapping pipeline, or you can configure an unconnected Lookup transformation to receive input from the result of an expression in another transformation.

The following table lists the differences between connected and unconnected lookups:

| Connected Lookup | Unconnected Lookup |
| --- | --- |
| Receives input values directly from the pipeline. | Receives input values from the result of a :LKP expression in another transformation. |
| Use a dynamic or static cache. | Use a static cache. |
| Cache includes the lookup source columns in the lookup condition and the lookup source columns that are output ports. | Cache includes all lookup/output ports in the lookup condition and the lookup/return port. |
| Can return multiple columns from the same row or insert into the dynamic lookup cache. | Designate one return port (R). Returns one column from each row. |
| If there is no match for the lookup condition, the Integration Service returns the default value for all output ports. If you configure dynamic caching, the Integration Service inserts rows into the cache or leaves it unchanged. | If there is no match for the lookup condition, the Integration Service returns NULL. |
| If there is a match for the lookup condition, the Integration Service returns the result of the lookup condition for all lookup/output ports. If you configure dynamic caching, the Integration Service either updates the row the in the cache or leaves the row unchanged. | If there is a match for the lookup condition, the Integration Service returns the result of the lookup condition into the return port. |
| Pass multiple output values to another transformation. Link lookup/output ports to another transformation. | Pass one output value to another transformation. The lookup/output/return port passes the value to the transformation calling :LKP expression. |
| Supports user-defined default values. | Does not support user-defined default values. |

**Connected Lookup Transformation**

The following steps describe how the Integration Service processes a connected Lookup transformation:

1. A connected Lookup transformation receives input values directly from another transformation in the pipeline.
2. For each input row, the Integration Service queries the lookup source or cache based on the lookup ports and the condition in the transformation.
3. If the transformation is uncached or uses a static cache, the Integration Service returns values from the lookup query.
4. If the transformation uses a dynamic cache, the Integration Service inserts the row into the cache when it does not find the row in the cache. When the Integration Service finds the row in the cache, it updates the row in the cache or leaves it unchanged. It flags the row as insert, update, or no change.
5. The Integration Service passes return values from the query to the next transformation.
6. If the transformation uses a dynamic cache, you can pass rows to a Filter or Router transformation to filter new rows to the target.

**Note:** This chapter discusses connected Lookup transformations unless otherwise specified.

**Unconnected Lookup Transformation**

An unconnected Lookup transformation receives input values from the result of a :LKP expression in another transformation. You can call the Lookup transformation more than once in a mapping.

A common use for unconnected Lookup transformations is to update slowly changing dimension tables.

The following steps describe the way the Integration Service processes an unconnected Lookup transformation:

1. An unconnected Lookup transformation receives input values from the result of a :LKP expression in another transformation, such as an Update Strategy transformation.
2. The Integration Service queries the lookup source or cache based on the lookup ports and condition in the transformation.
3. The Integration Service returns one value into the return port of the Lookup transformation.
4. The Lookup transformation passes the return value into the :LKP expression.

**Lookup Components**

Define the following components when you configure a Lookup transformation in a mapping:

* Lookup source
* Ports
* Properties
* Condition

**Lookup Source**

We use a flat file, relational table, or source qualifier for a lookup source. When you create a Lookup transformation, you can create the lookup source from the following locations:

* Relational source or target definition in the repository
* Flat file source or target definition in the repository
* Table or file that the Integration Service and PowerCenter Client machine can connect to
* Source qualifier definition in a mapping

The lookup table can be a single table, or you can join multiple tables in the same database using a lookup SQL override. The Integration Service queries the lookup table or an in-memory cache of the table for all incoming rows into the Lookup transformation.

The Integration Service can connect to a lookup table using ODBC or native drivers. Configure native drivers for optimal performance.

**Indexes and a Lookup Table**

If you have privileges to modify the database containing a lookup table, you can improve lookup initialization time by adding an index to the lookup table. You can improve performance for very large lookup tables. Since the Integration Service queries, sorts, and compares values in lookup columns, the index needs to include every column in a lookup condition.

You can improve performance by indexing the following types of lookup:

* **Cached lookups.** You can improve performance by indexing the columns in the lookup ORDER BY. The session log contains the ORDER BY clause.
* **Uncached lookups.** Because the Integration Service issues a SELECT statement for each row passing into the Lookup transformation, you can improve performance by indexing the columns in the lookup condition.

**Lookup Ports**

The Ports tab contains input and output ports. The Ports tab also includes lookup ports that represent columns of data to return from the lookup source. An unconnected Lookup transformation returns one column of data to the calling transformation in this port. An unconnected Lookup transformation has one return port.

| Ports | Type of Lookup | Description |
| --- | --- | --- |
| I | Connected  Unconnected | Input port. Create an input port for each lookup port you want to use in the lookup condition. You must have at least one input or input/output port in each Lookup transformation. |
| O | Connected  Unconnected | Output port. Create an output port for each lookup port you want to link to another transformation. You can designate both input and lookup ports as output ports. For connected lookups, you must have at least one output port. For unconnected lookups, select a lookup port as a return port (R) to pass a return value. |
| L | Connected  Unconnected | Lookup port. The Designer designates each column in the lookup source as a lookup (L) and output port (O). |
| R | Unconnected | Return port. Use only in unconnected Lookup transformations. Designates the column of data you want to return based on the lookup condition. You can designate one lookup port as the return port. |

The Lookup transformation also enables an associated expression property that you configure when you use a dynamic cache. The associated expression property contains the data to update the lookup cache. It can contain an expression to update the dynamic cache or it can contain an input port name.

Use the following guidelines to configure lookup ports:

* If you delete lookup ports from a flat file lookup, the session fails.
* You can delete lookup ports from a relational lookup if the mapping does not use the lookup port. This reduces the amount of memory the Integration Service needs to run the session.

**Lookup Properties**

Configure the lookup properties such as caching and multiple matches on the Lookup Properties tab. Configure the lookup condition or the SQL statements to query the lookup table. We can also change the Lookup table name.

When we create a mapping, we configure the properties for each Lookup transformation. When we create a session, we can override properties such as the index and the data cache size for each transformation.

The following table describes the Lookup transformation properties:

| Option | Lookup Type | Description |
| --- | --- | --- |
| Lookup SQL Override | Relational | Overrides the default SQL statement to query the lookup table.  Specifies the SQL statement you want the Integration Service to use for querying lookup values. Use with the lookup cache enabled. |
| Lookup Table Name | Pipeline  Relational | The name of the table or the source qualifier from which the transformation looks up and caches values. When you create the Lookup transformation, choose a source, target, or source qualifier as the lookup source. You can also import a table, view, or synonym from another database when you create the Lookup transformation.  If you enter a lookup SQL override, you do not need to enter the Lookup Table Name. |
| Lookup Source Filter | Relational | Restricts the lookups the Integration Service performs based on the value of data in any port in the Lookup transformation. Use with the lookup cache enabled. |
| Lookup Caching Enabled | Flat File  Pipeline  Relational | Indicates whether the Integration Service caches lookup values during the session.  When you enable lookup caching, the Integration Service queries the lookup source once, caches the values, and looks up values in the cache during the session. Caching the lookup values can improve session performance.  When you disable caching, each time a row passes into the transformation, the Integration Service issues a select statement to the lookup source for lookup values.  **Note:** The Integration Service always caches the flat file lookups and the pipeline lookups. |
| Lookup Policy on Multiple Match | Flat File  Pipeline  Relational | Determines which rows to return when the Lookup transformation finds multiple rows that match the lookup condition. Select one of the following values:   * Report Error. The Integration Service reports an error and does not return a row. * Use First Value. Returns the first row that matches the lookup condition. * Use Last Value. Return the last row that matches the lookup condition. * Use All Values. Return all matching rows. * Use Any Value. The Integration Service returns the first value that matches the lookup condition. It creates an index based on the key ports instead of all Lookup transformation ports. * Report Error. The Integration Service reports an error and does not return a row. If you do not enable the Output Old Value On Update option, the Lookup Policy On Multiple Match option is set to Report Error for dynamic lookups. |
| Lookup Condition | Flat File  Pipeline  Relational | Displays the lookup condition you set in the Condition tab. |
| Connection Information | Relational | Specifies the database that contains the lookup table. You can define the database in the mapping, session, or parameter file:   * Mapping. Select the connection object. You can also specify the database connection type. Type Relational:before the connection name if it is a relational connection. Type Application:before the connection name if it is an application connection. * Session. Use the $Source or $Target connection variable. If you use one of these variables, the lookup table must reside in the source or the target database. Specify the database connection in the session properties for each variable. * Parameter file. Use the session parameter $DBConnection*Name* or $AppConnection*Name*, and define it in the parameter file.   By default, the Designer specifies $Source if you choose a source table and $Target if you choose a target table when you create the Lookup transformation. You can override these values in the session properties.  The Integration Service fails the session if it cannot determine the type of database connection. |
| Source Type | Flat File  Pipeline  Relational | Indicates that the Lookup transformation reads values from a relational table, flat file, or source qualifier. |
| Tracing Level | Flat File  Pipeline  Relational | Sets the amount of detail included in the session log. |
| Lookup Cache Directory Name | Flat File  Pipeline  Relational | Specifies the directory used to build the lookup cache files when you configure the Lookup transformation to cache the lookup source. Also saves the persistent lookup cache files when you select the Lookup Persistent option.  By default, the Integration Service uses the $PMCacheDir directory configured for the Integration Service. |
| Lookup Cache Persistent | Flat File  Pipeline  Relational | Indicates whether the Integration Service uses a persistent lookup cache, which consists of at least two cache files. If a Lookup transformation is configured for a persistent lookup cache and persistent lookup cache files do not exist, the Integration Service creates the files during the session. Use with the lookup cache enabled. |
| Lookup Data Cache Size Lookup Index Cache Size | Flat File  Pipeline  Relational | Default is Auto. Indicates the maximum size the Integration Service allocates to the data cache and the index in memory. You can configure a numeric value, or you can configure the Integration Service to determine the cache size at run time. If you configure the Integration Service to determine the cache size, you can also configure a maximum amount of memory for the Integration Service to allocate to the cache.  If the Integration Service cannot allocate the configured amount of memory when initializing the session, it fails the session. When the Integration Service cannot store all the data cache data in memory, it pages to disk.  Use with the lookup cache enabled. |
| Dynamic Lookup Cache | Flat File  Pipeline  Relational | Indicates to use a dynamic lookup cache. Inserts or updates rows in the lookup cache as it passes rows to the target table.  Use with the lookup cache enabled. |
| Output Old Value On Update | Flat File  Pipeline  Relational | Use with dynamic caching enabled. When you enable this property, the Integration Service outputs old values out of the lookup/output ports. When the Integration Service updates a row in the cache, it outputs the value that existed in the lookup cache before it updated the row based on the input data. When the Integration Service inserts a row in the cache, it outputs null values.  When you disable this property, the Integration Service outputs the same values out of the lookup/output and input/output ports.  This property is enabled by default. |
| Update Dynamic Cache Condition | Flat File  Pipeline  Relational | An expression that indicates whether to update dynamic cache. Create an expression using lookup ports or input ports. The expression can contain input values or values in the lookup cache. The Integration Service updates the cache when the condition is true and the data exists in the cache. Use with dynamic caching enabled. Default is true. |
| Cache File Name Prefix | Flat File  Pipeline  Relational | Use with persistent lookup cache. Specifies the file name prefix to use with persistent lookup cache files. The Integration Service uses the file name prefix as the file name for the persistent cache files it saves to disk. Enter the prefix. Do not enter .idx or .dat.  You can enter a parameter or variable for the file name prefix. Use any parameter or variable type that you can define in the parameter file.  If the named persistent cache files exist, the Integration Service builds the memory cache from the files. If the named persistent cache files do not exist, the Integration Service rebuilds the persistent cache files. |
| Recache From Lookup Source | Flat File  Pipeline  Relational | Use with the lookup cache enabled. When selected, the Integration Service rebuilds the lookup cache from the lookup source when it first calls the Lookup transformation instance.  If you use a persistent lookup cache, it rebuilds the persistent cache files before using the cache. If you do not use a persistent lookup cache, it rebuilds the lookup cache in the memory before using the cache. |
| Insert Else Update | Flat File  Pipeline  Relational | Use with dynamic caching enabled. Applies to rows entering the Lookup transformation with the row type of insert. When enabled, the Integration Service inserts rows in the cache and updates existing rows When disabled, the Integration Service does not update existing rows. |
| Update Else Insert | Flat File  Pipeline  Relational | Use with dynamic caching enabled. Applies to rows entering the Lookup transformation with the row type of update.  When enabled, the Integration Service updates existing rows, and inserts a row if it is new. When disabled, the Integration Service does not insert new rows. |
| Datetime Format | Flat File | Click the Open button to select a datetime format. Define the format and the field width. Milliseconds, microseconds, or nanoseconds formats have a field width of 29.  If you do not select a datetime format for a port, you can enter any datetime format. Default is MM/DD/YYYY HH24:MI:SS. The Datetime format does not change the size of the port. |
| Thousand Separator | Flat File | If you do not define a thousand separator for a port, the Integration Service uses the properties defined here.  You can choose no separator, a comma, or a period. Default is no separator. |
| Decimal Separator | Flat File | If you do not define a decimal separator for a particular field in the lookup definition or on the Ports tab, the Integration Service uses the properties defined here.  You can choose a comma or a period decimal separator. Default is period. |
| Case-Sensitive String Comparison | Flat File  Pipeline | The Integration Service uses case sensitive string comparisons when performing lookups on string columns.  For relational lookups, the case sensitive comparison depends on the database support. |
| Null Ordering | Flat File  Pipeline | Determines how the Integration Service orders null values. You can choose to sort null values high or low. By default, the Integration Service sorts null values high. This overrides the Integration Service configuration to treat nulls in comparison operators as high, low, or null.  For relational lookups, null ordering depends on the database default value. |
| Sorted Input | Flat File  Pipeline | Indicates whether or not the lookup file data is in sorted order. This increases lookup performance for file lookups. If you enable sorted input, and the condition columns are not grouped, the Integration Service fails the session. If the condition columns are grouped,   but not sorted, the Integration Service processes the lookup as if you did not configure sorted input. |
| Lookup Source is Static | Flat File  Pipeline  Relational | The lookup source does not change in a session. |
| Pre-build Lookup Cache | Flat File  Pipeline  Relational | Allows the Integration Service to build the lookup cache before the Lookup transformation receives the data. The Integration Service can build multiple lookup cache files at the same time to improve performance.  You can configure this option in the mapping or the session. The Integration Service uses the session-level setting if you configure the Lookup transformation option as Auto.  Configure one of the following options:   * Auto. The Integration Service uses the value   configured in the session. * Always allowed. The Integration Service can build the lookup cache before the Lookup transformation receives the first source row. The Integration Service creates an additional pipeline to build the cache. * Always disallowed. The Integration Service cannot build the lookup cache before the Lookup transformation receives the first row.   You must configure the number of pipelines that the Integration Service can build concurrently. Configure the Additional Concurrent Pipelines for Lookup Cache Creation session property. The Integration Service can pre-build lookup cache if this property is greater than zero. |
| Subsecond Precision | Relational | Specifies the subsecond precision for datetime ports.  For relational lookups, you can change the precision for databases that have an editable scale for datetime data. You can change subsecond precision for Oracle Timestamp, Informix Datetime, and Teradata Timestamp datatypes.  Enter a positive integer value from 0 to 9. Default is 6 microseconds. If you enable pushdown optimization, the database returns the complete datetime value, regardless of the subsecond precision setting. |

**Configuring Lookup Properties in a Session**

[When you configure a session, you can configure lookup properties that are unique to sessions:](https://www.blogger.com/blogger.g?blogID=3888677960781784215)

**Flat file lookups:** Configure lookup location information, such as the source file directory, file name, and the file type.

**Relational lookups:** You can define $Source and $Target variables in the session properties. You can also override connection information to use the $DBConnection*Name* or $AppConnection*Name* session parameter.

**Pipeline lookups:** Configure the lookup source file properties such as the source file directory, file name, and the file type. If the source is a relational table or application source, configure the connection information.

**Configuring Flat File Lookups in a Session**

When you configure a flat file lookup in a session, configure the lookup source file properties on the Transformation View of the Mapping tab. Choose the Lookup transformation and configure the flat file properties in the session properties for the transformation.

The following table describes the session properties you configure for flat file lookups:

* Lookup Source File Directory will be **$PMLookupFileDir**/<your\_project\_foldername> (default **$PMLookupFileDir**)
* Lookup Source Filename you can either given the lookup file name or use parameter $LookupFile*Name ($Lookup is prefix to lookup file name).*
* Lookup Source Filetype will Indicates whether the lookup source file contains the source data or a list of files with the same file properties. Choose Direct if the lookup source file contains the source data. Choose Indirect if the lookup source file contains a list of files.

**Configuring Relational Lookups in a Session**

When you configure a relational lookup in a session, configure the connection for the lookup database on the Transformation View of the Mapping tab. Choose the Lookup transformation and configure the connection in the session properties for the transformation.

Choose from the following options to configure a connection for a relational Lookup transformation:

* Choose a relational or application connection.
* Configure a database connection using the $Source or $Target connection variable.
* Configure the session parameter **$DBConnection*Name*** or **$AppConnection*Name***, and define the session parameter in a parameter file.

**Configuring Pipeline Lookups in a Session**

When you configure a pipeline Lookup in a session, configure the location of lookup source file or the connection for the lookup table on the Sources node of the Mapping tab. Choose the Source Qualifier that represents the lookup source.

**Lookup Query**

The Integration Service queries the lookup based on the ports and properties you configure in the Lookup transformation. The Integration Service runs a default SQL statement when the first row enters the Lookup transformation. If you use a relational lookup or a pipeline lookup against a relational table, you can customize the default query with the Lookup SQL Override property.

You can restrict the rows that a Lookup transformation retrieves from the source when it builds the lookup cache. Configure the Lookup Source Filter.

If you configure both the Lookup SQL Override and the Lookup Source Filter properties, the Integration Service ignores the Lookup Source Filter property.

**Default Lookup Query**

* The default lookup query contains the following statements:
* **SELECT.** The SELECT statement includes all the lookup ports in the mapping. You can view the SELECT statement by generating SQL using the Lookup SQL Override property. Do not add or delete any columns from the default SQL statement.
* **ORDER BY.** The ORDER BY clause orders the columns in the same order they appear in the Lookup transformation. The Integration Service generates the ORDER BY clause. You cannot view this when you generate the default SQL using the Lookup SQL Override property.

**Overriding the ORDER BY Clause**

By default, the Integration Service generates an ORDER BY clause for a cached lookup. The ORDER BY clause contains all lookup ports. To increase performance, you can suppress the default ORDER BY clause and enter an override ORDER BY with fewer columns.

**Note**: If you use pushdown optimization, you cannot override the ORDER BY clause or suppress the generated ORDER BY clause with a comment notation.

The Integration Service always generates an ORDER BY clause, even if you enter one in the override. Place two dashes ‘--’ after the ORDER BY override to suppress the generated ORDER BY clause. For example, a Lookup transformation uses the following lookup condition:

ITEM\_ID = IN\_ITEM\_ID PRICE <= IN\_PRICE

The Lookup transformation includes three lookup ports used in the mapping, ITEM\_ID, ITEM\_NAME, and PRICE. When you enter the ORDER BY clause, enter the columns in the same order as the ports in the lookup condition. You must also enclose all database reserved words in quotes. Enter the following lookup query in the lookup SQL override:

SELECT ITEMS\_DIM.ITEM\_NAME, ITEMS\_DIM.PRICE, ITEMS\_DIM.ITEM\_ID FROM ITEMS\_DIM ORDER BY ITEMS\_DIM.ITEM\_ID, ITEMS\_DIM.PRICE --

**Lookup Condition**

The Integration Service finds data in the lookup source with a lookup condition. The lookup condition is similar to the WHERE clause in an SQL query. When you configure a lookup condition in a Lookup transformation, you compare the value of one or more columns in the source data with values in the lookup source or cache.

Use the following guidelines when you enter a condition for a Lookup transformation:

* The datatypes for the columns in a lookup condition must match.
* You must enter a lookup condition in all Lookup transformations.
* Use one input port for each lookup port in the lookup condition. Use the same input port in more than one condition in a transformation.
* When you enter multiple conditions, the Integration Service evaluates each condition as an AND, not an OR. The Integration Service returns rows that match all the conditions you configure.
* If you include multiple conditions, enter the conditions in the following order to optimize lookup performance:
  + Equal to (=)
  + Less than (<), greater than (>), less than or equal to (<=), greater than or equal to (>=)
  + Not equal to (!=)
* The Integration Service matches null values. For example, if an input lookup condition column is NULL, the Integration Service evaluates the NULL equal to a NULL in the lookup.
* If you configure a flat file lookup for sorted input, the Integration Service fails the session if the condition columns are not grouped. If the columns are grouped, but not sorted, the Integration Service processes the lookup as if you did not configure sorted input.

The Integration Service processes lookup matches differently depending on whether you configure the transformation for a dynamic cache or an uncached or static cache.

**Uncached or Static Cache**

Use the following guidelines when you configure a Lookup transformation that has a static lookup cache or an uncached lookup source:

* Use the following operators when you create the lookup condition:

                   =, >, <, >=, <=, !=

If you include more than one lookup condition, place the conditions in the following order to optimize lookup performance:

* Equal to (=)
* Less than (<), greater than (>), less than or equal to (<=), greater than or equal to (>=)
* Not equal to (!=)

          For example, create the following lookup condition:

          ITEM\_ID = IN\_ITEM\_ID PRICE <= IN\_PRICE

* The input value must meet all conditions for the lookup to return a value.

The condition can match equivalent values or supply a threshold condition. For example, you might look for customers who do not live in California, or employees whose salary is greater than $30,000. Depending on the nature of the source and condition, the lookup might return multiple values.

**Dynamic Cache**

If you configure a Lookup transformation to use a dynamic cache, you can use only the equality operator (=) in the lookup condition.

**Handling Multiple Matches**

The Lookup transformation finds values based on the condition you configure in the transformation. If the lookup condition is not based on a unique key, or if the lookup source is denormalized, the Integration Service might find multiple matches in the lookup source or the lookup cache.

You can configure a Lookup transformation to handle multiple matches in the following ways:

* **Use the first matching value, or use the last matching value.** We can configure the transformation to return the first matching value or the last matching value. The first and last values are the first value and last value found in the lookup cache that match the lookup condition. When you cache the lookup source, the Integration Service generates an ORDER BY clause for each column in the lookup cache to determine the first and last row in the cache. The Integration Service then sorts each lookup source column in ascending order.

The Integration Service sorts numeric columns in ascending numeric order such as 0 to 10. It sorts date/time columns from January to December and from the first of the month to the end of the month. The Integration Service sorts string columns based on the sort order configured for the session.

* **Use any matching value**. We can configure the Lookup transformation to return any value that matches the lookup condition. When we configure the Lookup transformation to return any matching value, the transformation returns the first value that matches the lookup condition. The transformation creates an index based on the key ports instead of all Lookup transformation ports. When you use any matching value, performance can improve because the process of indexing rows is simpler.
* **Use all values**. The Lookup transformation returns all matching rows. To use this option, you must configure the Lookup transformation to return all matches when you create the transformation. The transformation becomes an active transformation. You cannot change the mode between passive and active after you create the transformation.
* **Return an error.** When the Lookup transformation uses a static cache or no cache, the Integration Service marks the row as an error. The Lookup transformation writes the row to the session log by default, and increases the error count by one. When the Lookup transformation has a dynamic cache, the Integration Service fails the session when it encounters multiple matches. The session fails while the Integration Service is caching the lookup table or looking up the duplicate key values. Also, if you configure the Lookup transformation to output old values on updates, the Lookup transformation returns an error when it encounters multiple matches. The transformation creates an index based on the key ports instead of all Lookup transformation ports.

**Lookup Caches**

We can configure a Lookup transformation to cache the lookup file or table. The Integration Service builds a cache in memory when it processes the first row of data in a cached Lookup transformation. It allocates memory for the cache based on the amount you configure in the transformation or session properties. The Integration Service stores condition values in the index cache and output values in the data cache. The Integration Service queries the cache for each row that enters the transformation.

The Integration Service also creates cache files by default in the $PMCacheDir. If the data does not fit in the memory cache, the Integration Service stores the overflow values in the cache files. When the session completes, the Integration Service releases cache memory and deletes the cache files unless you configure the Lookup transformation to use a persistent cache.

When configuring a lookup cache, you can configure the following options:

* Persistent cache
* Recache from lookup source
* Static cache
* Dynamic cache
* Shared cache
* Pre-build lookup cache

**Note:** You can use a dynamic cache for relational or flat file lookups.

**Rules and Guidelines for Returning Multiple Rows**

Use the following rules and guidelines when you configure the Lookup transformation to return multiple rows:

* The Integration Service caches all rows from the lookup source for cached lookups.
* You can configure an SQL override for a cached or uncached lookup that returns multiple rows.
* You cannot enable dynamic cache for a Lookup transformation that returns multiple rows.
* You cannot return multiple rows from an unconnected Lookup transformation.
* You can configure multiple Lookup transformations to share a named cache if the Lookup transformations have matching caching lookup on multiple match policies.
* An Lookup transformation that returns multiple rows cannot share a cache with a Lookup transformation that returns one matching row for each input row.

**Lookup Caches Overview**

We can configure a Lookup transformation to cache the lookup table. The Integration Service builds a cache in memory when it processes the first row of data in a cached Lookup transformation. It allocates memory for the cache based on the amount you configure in the transformation or session properties. The Integration Service stores condition values in the index cache and output values in the data cache. The Integration Service queries the cache for each row that enters the transformation.

The Integration Service also creates cache files by default in the $PMCacheDir. If the data does not fit in the memory cache, the Integration Service stores the overflow values in the cache files. When the session completes, the Integration Service releases cache memory and deletes the cache files unless you configure the Lookup transformation to use a persistent cache.

If you use a flat file or pipeline lookup, the Integration Service always caches the lookup source. If you configure a flat file lookup for sorted input, the Integration Service cannot cache the lookup if the condition columns are not grouped. If the columns are grouped, but not sorted, the Integration Service processes the lookup as if you did not configure sorted input.

When you configure a lookup cache, you can configure the following cache settings:

* **Building caches:** We can configure the session to build caches sequentially or concurrently. When you build sequential caches, the Integration Service creates caches as the source rows enter the Lookup transformation. When you configure the session to build concurrent caches, the Integration Service does not wait for the first row to enter the Lookup transformation before it creates caches. Instead, it builds multiple caches concurrently.
* **Persistent cache:** We can save the lookup cache files and reuse them the next time the Integration Service processes a Lookup transformation configured to use the cache.
* **Recache from source:** If the persistent cache is not synchronized with the lookup table, you can configure the Lookup transformation to rebuild the lookup cache.
* **Static cache:** We can configure a static, or read-only, cache for any lookup source. By default, the Integration Service creates a static cache. It caches the lookup file or table and looks up values in the cache for each row that comes into the transformation. When the lookup condition is true, the Integration Service returns a value from the lookup cache. The Integration Service does not update the cache while it processes the Lookup transformation.
* **Dynamic cache:** To cache a table, flat file, or source definition and update the cache, configure a Lookup transformation with dynamic cache. The Integration Service dynamically inserts or updates data in the lookup cache and passes the data to the target. The dynamic cache is synchronized with the target.
* **Shared cache:** We can share the lookup cache between multiple transformations. We can share an unnamed cache between transformations in the same mapping. We can share a named cache between transformations in the same or different mappings. Lookup transformations can share unnamed static caches within the same target load order group if the cache sharing rules match. Lookup transformations cannot share dynamic cache within the same target load order group.

When you do not configure the Lookup transformation for caching, the Integration Service queries the lookup table for each input row. The result of the Lookup query and processing is the same, whether or not you cache the lookup table. However, using a lookup cache can increase session performance. Optimize performance by caching the lookup table when the source table is large.

**Note:** The Integration Service uses the same transformation logic to process a Lookup transformation whether you configure it to use a static cache or no cache. However, when you configure the transformation to use no cache, the Integration Service queries the lookup table instead of the lookup cache.

**Cache Comparison**

The following table compares the differences between an uncached lookup, a static cache, and a dynamic cache:

| Uncached | Static Cache | Dynamic Cache |
| --- | --- | --- |
| We cannot insert or update the cache. | We cannot insert or update the cache. | We can insert or update rows in the cache as you pass rows to the target. |
| We cannot use a flat file or pipeline lookup. | Use a relational, flat file, or pipeline lookup. | Use a relational, flat file, or Source Qualifier lookup. |
| When the condition is true, the Integration Service returns a value from the lookup   table or cache.  When the condition is not true, the Integration Service returns the default value for connected transformations and NULL for unconnected transformations. | When the condition is true, the Integration Service returns a value from the lookup   table or cache.  When the condition is not true, the Integration Service returns the default value for connected transformations and NULL for unconnected transformations. | When the condition is true, the Integration Service either updates rows in the cache or leaves the cache unchanged, depending on the row type. This indicates that the row is in the cache and target table. You can pass updated rows to a target.  When the condition is not true, the Integration Service either inserts rows into the cache or leaves the cache unchanged, depending on the row type. This indicates that the row is not in the cache or target. You can pass inserted rows to a target table. |

**Building Connected Lookup Caches**

The Integration Service can build lookup caches for connected Lookup transformations in the following ways:

* **Sequential caches**. The Integration Service builds lookup caches sequentially. The Integration Service builds the cache in memory when it processes the first row of the data in a cached lookup transformation.
* **Concurrent caches**. The Integration Service builds lookup caches concurrently. It does not need to wait for data to reach the Lookup transformation.

**Note:** The Integration Service builds caches for unconnected Lookup transformations sequentially regardless of how you configure cache building. If you configure the session to build concurrent caches for an unconnected Lookup transformation, the Integration Service ignores this setting and builds unconnected Lookup transformation caches sequentially.

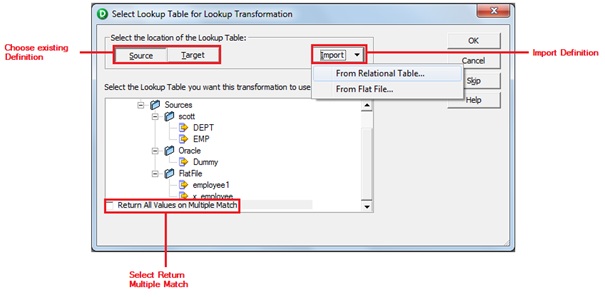
**How to Select/Import Lookup table?**

We can select source/target definition as lookup, if it already exist in the repository. If not we can import it from relational or Flat File.

1. Click on Lookup Icon highlighted below:

[https://2.bp.blogspot.com/-iQbJHAW5P2s/UW65aZSuAQI/AAAAAAAAAgk/P4511G25WcM/s1600/lkp1.jpg](https://2.bp.blogspot.com/-iQbJHAW5P2s/UW65aZSuAQI/AAAAAAAAAgk/P4511G25WcM/s1600/lkp1.jpg)

2. Click on Workspace

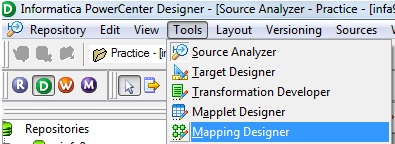
[](https://1.bp.blogspot.com/-W-sJt157FpQ/UW65aeRM8MI/AAAAAAAAAgo/pMwsFwBq-qU/s1600/lkp2.jpg)

**1. Sample Mapping wtih Connected Lookup:**

Create a mapping using EMP table and do lookup on DEPT table to get DNAME and LOC using DEPTNO.

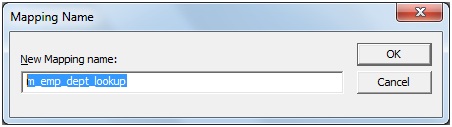
1.Connect and Open the folder if not already opened.

2. Select Tools --> Mapping Designer

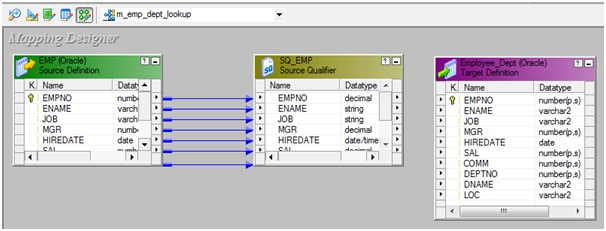
[](https://1.bp.blogspot.com/-Z9Xijhdr7ec/UW66SwoivbI/AAAAAAAAAjM/FjE0qpdyKrQ/s1600/sq-1.jpg)

3. Select Mappings --> Create

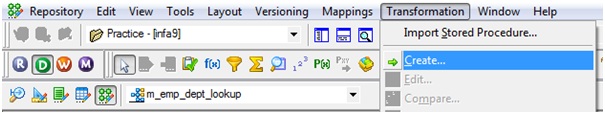
It will pop-up "Mapping Name". Enter the mapping name of your choice" and Click on 'OK'. Example: m\_emp\_dept\_lookup

[](https://4.bp.blogspot.com/-7GfVOVfef-U/UW650rPja6I/AAAAAAAAAg0/VJh1huVOW2s/s1600/lkp3.jpg)

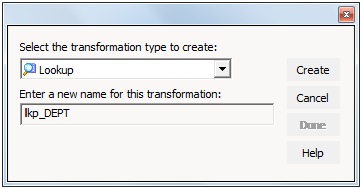
4. Drag the Source and Target definitions into workspace if they are already exist.[If not click here to know how to create or import Table definitions.](http://gowtham-informatica-reference.blogspot.in/2013/04/import-relation-definitions.html)

[](https://4.bp.blogspot.com/-7_a1LuG1WEk/UW658UGDLkI/AAAAAAAAAg8/msUTTkU-zg8/s1600/lkp4.jpg)

5. Select 'Transformation' from Menu --> Create

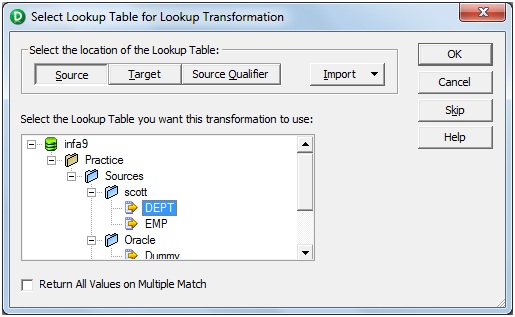
[](https://4.bp.blogspot.com/-YYhTEVbsE7I/UW66O-spV5I/AAAAAAAAAiU/Xpi3gVaT1BI/s1600/lkp5.jpg)

a) That will appear you 'Select the transformation type to create:'

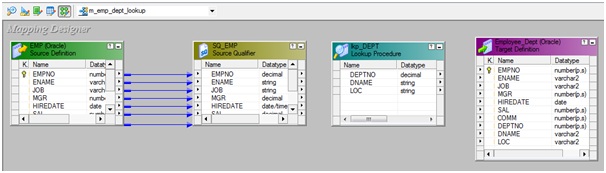
[](https://2.bp.blogspot.com/-GkyyMxBkfGw/UW66PSIstpI/AAAAAAAAAio/3aR-CYngwxY/s1600/lkp6.jpg)

b) Select 'Lookup' from drop down and 'Enter a new name for this transformation:' as "lkp\_DEPT"

c) Click 'Create' then it will pop-up 'Select Lookup table for Lookup Transformation'

[](https://3.bp.blogspot.com/-7K4eMbsHHb8/UW66PmrYIiI/AAAAAAAAAig/WBv4gbRaxAw/s1600/lkp7.jpg)

d) Select the Source definition: DEPT then click 'OK' and Done.

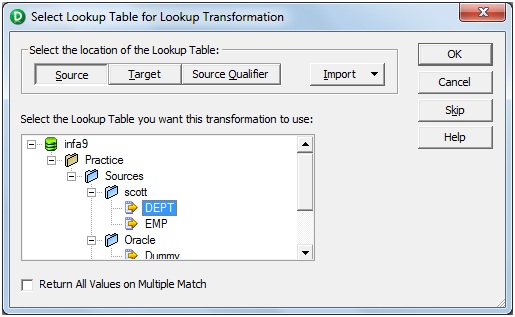
[](https://4.bp.blogspot.com/-065U7GINnV0/UW66G--cHuI/AAAAAAAAAhM/Krl8QU37BMc/s1600/lkp10.jpg)

**OR**

 a) Click on Lookup Transformation icon marked below in below snapshot.

[https://4.bp.blogspot.com/-iQbJHAW5P2s/UW65aZSuAQI/AAAAAAAAAgs/WRqhfJVTI6c/s1600/lkp1.jpg](https://4.bp.blogspot.com/-iQbJHAW5P2s/UW65aZSuAQI/AAAAAAAAAgs/WRqhfJVTI6c/s1600/lkp1.jpg)

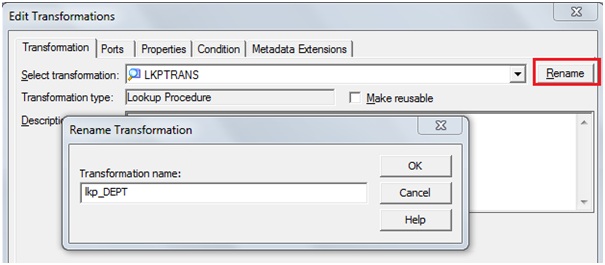
b) Click in the workspace in Mapping Designer.

[](https://3.bp.blogspot.com/-7K4eMbsHHb8/UW66PmrYIiI/AAAAAAAAAig/WBv4gbRaxAw/s1600/lkp7.jpg)

c) Select Lookup table location: 'Source'

d) Select Definition: DEPT and Click 'OK'

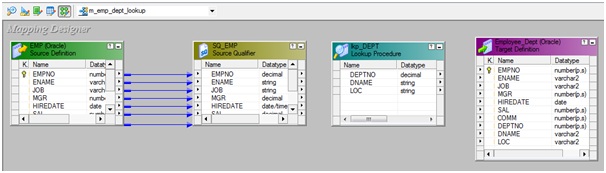
e) Select LKPTRANS in workspace and Right Click --> Edit.

[](https://3.bp.blogspot.com/-ChSJZuNLa5A/UW66QCpjjuI/AAAAAAAAAiw/txXwXasRYCY/s1600/lkp9.jpg)

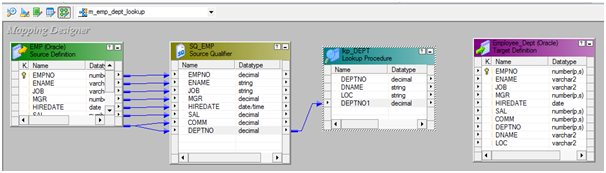
f) In Transformation tab --> Click on 'Rename' highlighted above which will pop-up 'Rename Transformation'. Enter the Transformation Name: "lkp\_DEPT"

g) Click on 'OK'

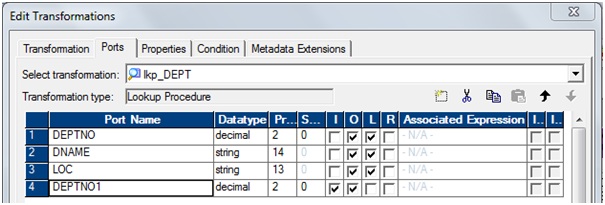
h) Click on 'Apply' and 'OK'.

[](https://1.bp.blogspot.com/-uznt23GNd0A/UW66Po-2OpI/AAAAAAAAAik/vePgqx36sBM/s1600/lkp8.jpg)

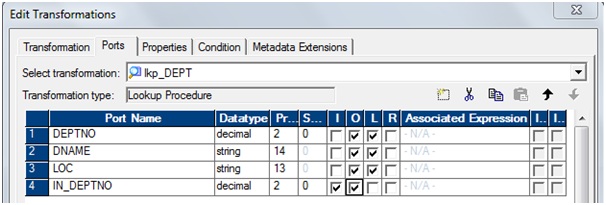
6). Drag required (all) ports from 'SQ\_EMP' Source Qualifier to 'lkp\_DEPT' lookup Transformation.

[](https://4.bp.blogspot.com/-SndLkfOFOdk/UW66G_YPntI/AAAAAAAAAhI/K_VxXmKUZKY/s1600/lkp11.png)

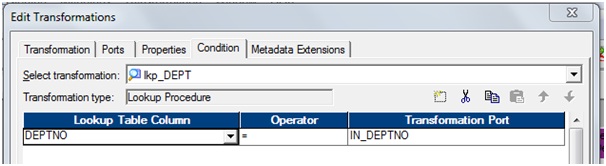
7) Select 'lkp\_DEPT' Lookup Transformation and Right Click --> Edit --> Ports Tab.

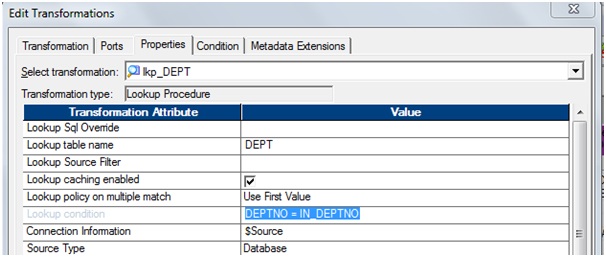
[](https://3.bp.blogspot.com/-0enZWFAxevM/UW66G2CfYfI/AAAAAAAAAhE/Y-7HxZuWwFs/s1600/lkp12.jpg)

a)  Rename 'DEPTNO1' to 'IN\_DEPTNO'

[](https://1.bp.blogspot.com/-uw8c6qrilV0/UW66IchekdI/AAAAAAAAAhU/E5gTjaNgBY8/s1600/lkp13.jpg)

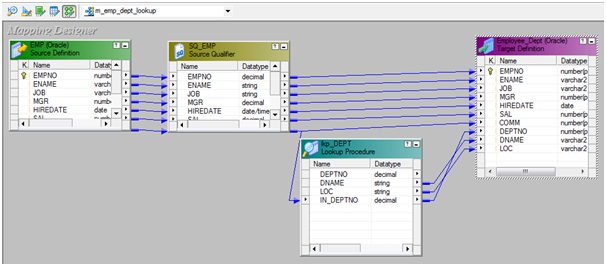
b) Go to 'Condition' tab: Add a port and then match condition below.

[](https://2.bp.blogspot.com/-xbYoKhvrNXE/UW66Ingy1WI/AAAAAAAAAhY/izd6RHaFK1g/s1600/lkp14.jpg)  
 c) Go to 'Properties' tab: Check the Lookup Condition'

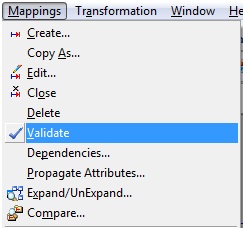
[](https://4.bp.blogspot.com/-mafYqDR8yvk/UW66Igzj8pI/AAAAAAAAAhg/R40PlF5yMa4/s1600/lkp15.jpg)

d) Click on 'Apply' and Click on 'OK'.

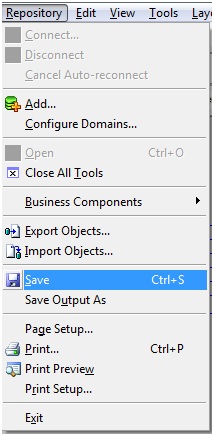
8) Drag all the required ports from SQ\_EMP and lkp\_DEPT to Employee\_Dept.

[](https://3.bp.blogspot.com/-O22w5ERgv6U/UW66IwN16II/AAAAAAAAAhc/TrzH3EG1rTg/s1600/lkp16.jpg)

9) Select 'Mapping' from Menu --> Validate.

[](https://3.bp.blogspot.com/-vjnvu2oJSms/UW66SwhRi5I/AAAAAAAAAjE/Ao7IR-1mmKo/s1600/sq-14.jpg)

10) Select 'Repository' from Menu --> Save.

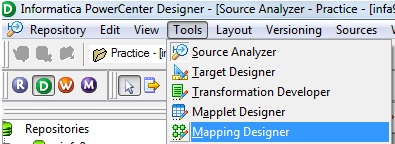
[](https://4.bp.blogspot.com/-5HQOxixhPoE/UW66S_n0RQI/AAAAAAAAAjI/yUUSkQeDhss/s1600/sq-15.jpg)

**2. Sample Mapping wtih Unconnected Looup:**

Create a mapping using EMP table and do lookup on DEPT table to get DNAME using DEPTNO. (Unconnected lookup will return only single value for each row).

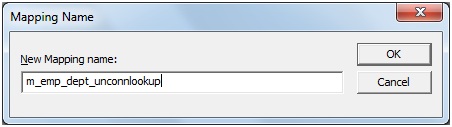
1.Connect and Open the folder if not already opened.

2. Select Tools --> Mapping Designer

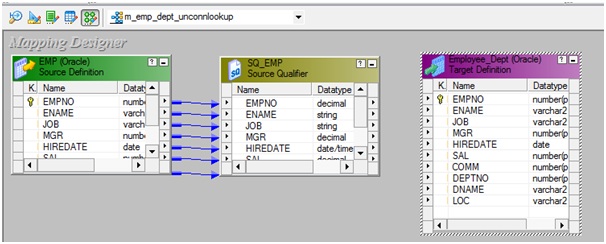
[](https://1.bp.blogspot.com/-Z9Xijhdr7ec/UW66SwoivbI/AAAAAAAAAjM/FjE0qpdyKrQ/s1600/sq-1.jpg)

3. Select Mappings --> Create

It will pop-up "Mapping Name". Enter the mapping name of your choice" and Click on 'OK'. Example: m\_emp\_dept\_unconnlookup

[](https://3.bp.blogspot.com/-sI0Iv-_mKak/UW66LbV_16I/AAAAAAAAAho/l5qQmTg7gGc/s1600/lkp17.jpg)

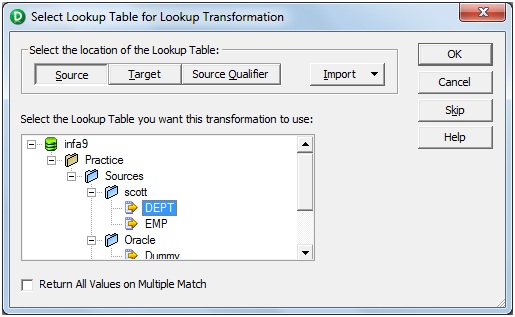
4. Drag the Source and Target definitions into workspace if they are already exist. If not click here to know how to create or import Table definitions.

[](https://4.bp.blogspot.com/-uEfPnzBzhv0/UW66LiesE5I/AAAAAAAAAhw/Ul2pdIqICUM/s1600/lkp18.jpg)

5. Select 'Transformation' from transformations menu

a) Click on Lookup Transformation icon marked below in below snapshot.  
 [https://4.bp.blogspot.com/-iQbJHAW5P2s/UW65aZSuAQI/AAAAAAAAAgs/WRqhfJVTI6c/s1600/lkp1.jpg](https://4.bp.blogspot.com/-iQbJHAW5P2s/UW65aZSuAQI/AAAAAAAAAgs/WRqhfJVTI6c/s1600/lkp1.jpg)

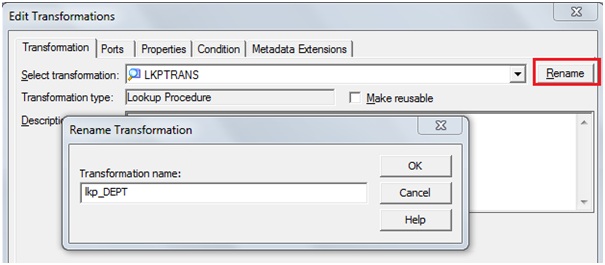
b) Click in the workspace in Mapping Designer.

[](https://3.bp.blogspot.com/-7K4eMbsHHb8/UW66PmrYIiI/AAAAAAAAAig/WBv4gbRaxAw/s1600/lkp7.jpg)

c) Select Lookup table location: 'Source'

d) Select Definition: DEPT and Click 'OK'

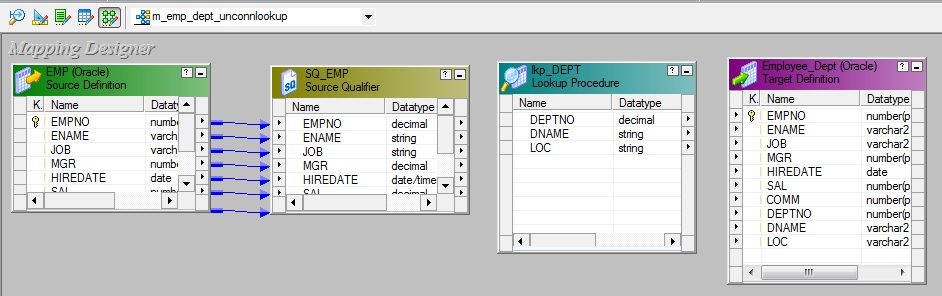
e) Select LKPTRANS in workspace and Right Click --> Edit.

[](https://3.bp.blogspot.com/-ChSJZuNLa5A/UW66QCpjjuI/AAAAAAAAAiw/txXwXasRYCY/s1600/lkp9.jpg)

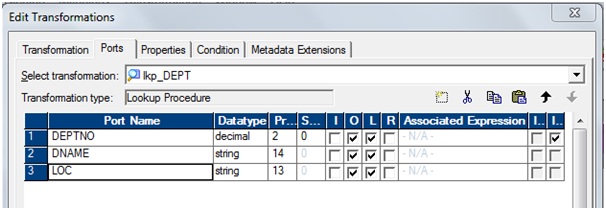
f) In Transformation tab --> Click on 'Rename' highlighted above which will pop-up 'Rename Transformation'. Enter the Transformation Name: "lkp\_DEPT"

g) Click on 'OK'

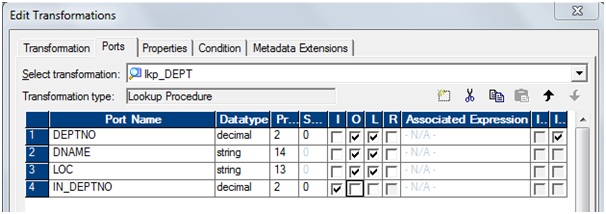
h) Click on 'Apply' and 'OK'.

[](https://1.bp.blogspot.com/-LoK3Cprijh8/UW66LxyO1RI/AAAAAAAAAhs/Y6LtNym5up4/s1600/lkp19.jpg)

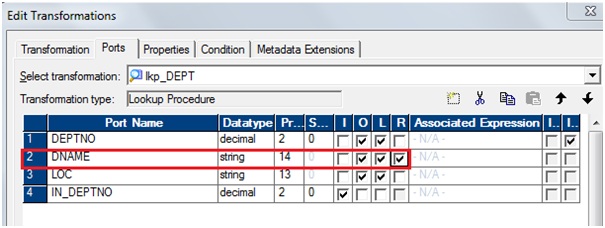
 6) Select 'lkp\_DEPT' Lookup Transformation and Right Click --> Edit --> Ports Tab.

[](https://1.bp.blogspot.com/-ro963Oh52_w/UW66MCD7l9I/AAAAAAAAAh0/81XNfwcCjxQ/s1600/lkp20.jpg)

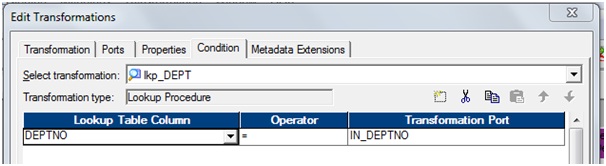
a) Add a port 'IN\_DEPTNO'

[](https://2.bp.blogspot.com/-ZSZ0PeKawIE/UW66McjcJcI/AAAAAAAAAh8/DGBQ53weNFs/s1600/lkp21.jpg)

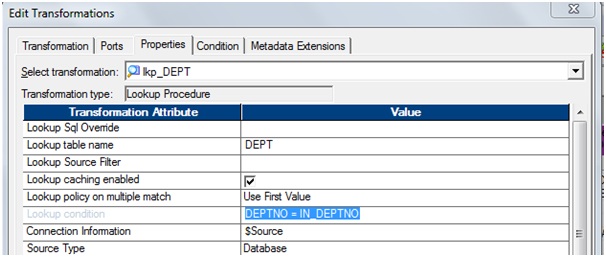
b) Select 'DNAME' as return port.

[](https://3.bp.blogspot.com/-8HvSi8NVMBo/UW66MbHkmrI/AAAAAAAAAh4/nb6PJnYVlaA/s1600/lkp22.jpg)

 c) Go to 'Condition' tab: Add a port and then match condition below

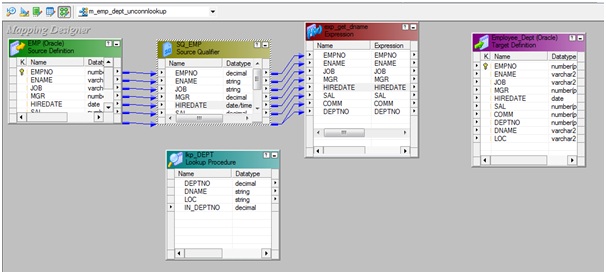
[](https://2.bp.blogspot.com/-xbYoKhvrNXE/UW66Ingy1WI/AAAAAAAAAhY/izd6RHaFK1g/s1600/lkp14.jpg)

d) Go to 'Properties' tab: Check the Lookup Condition'

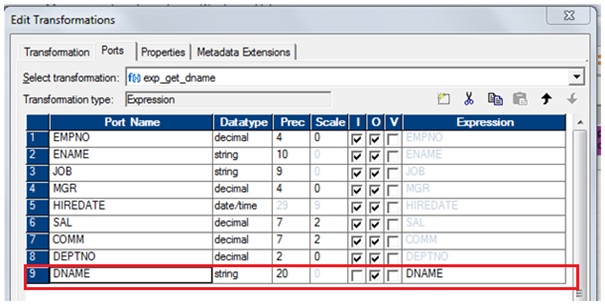
[](https://4.bp.blogspot.com/-mafYqDR8yvk/UW66Igzj8pI/AAAAAAAAAhg/R40PlF5yMa4/s1600/lkp15.jpg)

e) Click on 'Apply' and Click on 'OK'.

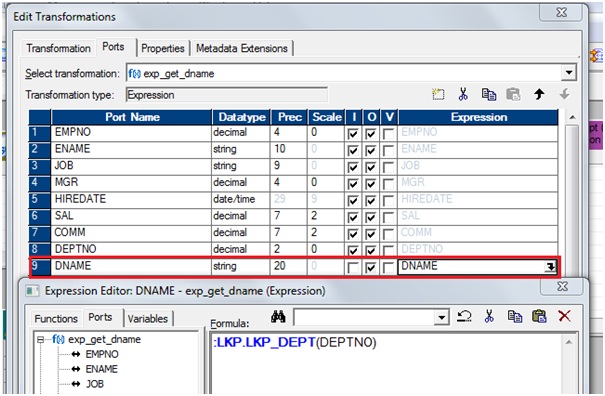
7) Create Expression transformation 'exp\_get\_dname' and drag all the required ports from SQ\_EMP.

[](https://4.bp.blogspot.com/-T5uGO0lF-Ck/UW66NxmmTyI/AAAAAAAAAiA/6jhl1qV_awY/s1600/lkp23.jpg)

8) Select 'exp\_get\_dname' expression, edit it and add Output port 'DNAME'.

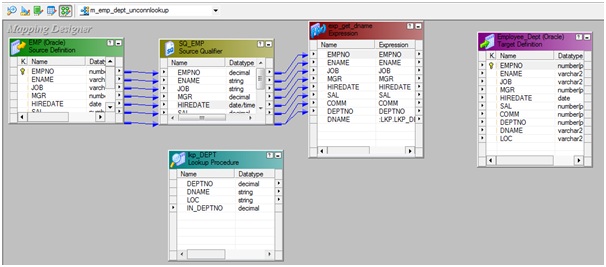
[](https://2.bp.blogspot.com/-BBX4nQExM_k/UW66OKbDDdI/AAAAAAAAAiQ/a-9h4HRBAUg/s1600/lkp24.jpg)

9) Click 'DNAME' browse editor to call unconnected lookup.

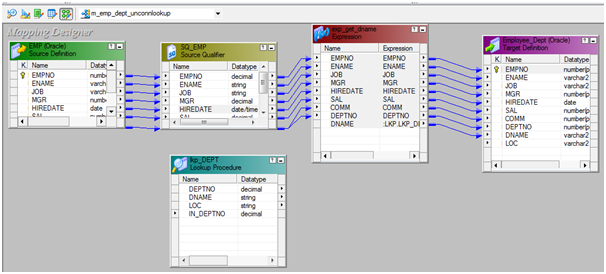
[](https://4.bp.blogspot.com/-M22TbEchmMc/UW66OLMioUI/AAAAAAAAAiI/22XqYMH6IAs/s1600/lkp25.jpg)

Note: Unconnected lookup can be called "**:LKP**"<unconnected\_lookupname> and sequence input arguments.

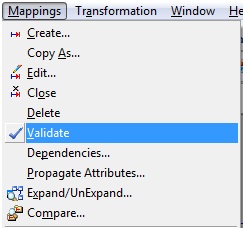
10) Click 'OK' then 'Apply' and Click 'OK'.

[](https://2.bp.blogspot.com/-8AU-tHIaHTs/UW66OQdZOsI/AAAAAAAAAiM/zAH7YTCj1fU/s1600/lkp26.jpg)

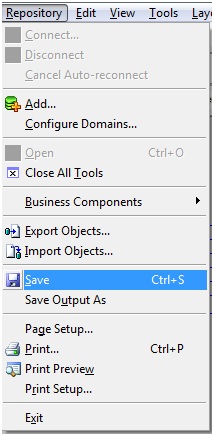
11) Drag required port from 'exp\_get\_dname' expression to 'Employee\_Dept' target table.

[](https://3.bp.blogspot.com/-dOJD_PTWgiQ/UW66O4N2XaI/AAAAAAAAAiY/ANVbqMFNkDw/s1600/lkp27.png)

 12) Select 'Mapping' from Menu --> Validate.

[](https://3.bp.blogspot.com/-vjnvu2oJSms/UW66SwhRi5I/AAAAAAAAAjE/Ao7IR-1mmKo/s1600/sq-14.jpg)

 13) Select 'Repository' from Menu --> Save.

[](https://4.bp.blogspot.com/-5HQOxixhPoE/UW66S_n0RQI/AAAAAAAAAjI/yUUSkQeDhss/s1600/sq-15.jpg)