

Conversion guide for oracle External Table to DB2 external table:

Required steps to migrate oracle external table to Db2 11.5 external table:

- Change create table clause to create external table
- Add column definition and check if there is any datatype that is not supported on db2
- Change ORGANIZATION EXTERNAL clause to USING clause
- Specify fully qualified location in DATAOBJECT and ensure path defined in EXTBL_LOCATION
- Map options describing delimited file to equivalent Db2 options. In some cases, options may be combined or provided through other means e.g. DEFAULT DIRECTORY

Important option mappings:

- **Type:** This is oracle specific option and not supported on db2.
- **DEFAULT DIRECTORY:** This is oracle specific option. Db2 requires the file reference to be fully qualified.
- **LOCATION:** mapped to DATAOBJECT option.
- **TERMINATED BY:** Mapped to DELIMITER.
- Format RECORDS FIXED, RECORDS DELIMITED are supported on DB2
- **REJECT LIMIT** mapped to MAX_ERRORS

RECORDS FIXED FORMAT:

DB2 Syntax: FORMAT '**FIXED**'

- Files in fixed-length format use ordinal positions, which are offsets, to identify where fields are within the record.
- An end-of-record delimiter is required even for the last record
- The locations of delimiters are fixed and specified in the layout definition because the fields are fixed in size
- To load fixed-format data into the database, you must define the target data type for the fields and the locations within the record.
- You do not have to load all fields in a fixed-length format file.

The following parameters apply when the FORMAT option of the external table is set to FIXED

LAYOUT - A layout is an ordered collection of zone or field definitions.

- **USE TYPE** - Indicates whether a zone is a normal data zone, a reference zone, or a filler zone
- **NAME** - The name of the zone
- **TYPE** - Defines the type of the zone
- **STYLE** - Defines the zone representation
- **LENGTH** - Specified as bytes or characters followed by the number or the internal reference to the reference zone
- **NULLIF** - Definition of the zone NULLESS attribute

RECORDLENGTH - Specifies the length of the entire record.

RECORDS DELIMITED FORMAT:

DB2 Syntax: FORMAT 'TEXT'

- The data to be loaded or unloaded is in text-delimited format
- This is the default.
- **RECORDS DELIMITED BY** mapped to RECORD_DELIMITER
- **FIELDS TERMINATED BY** mapped to DELIMITER
- BADFILE/ LOGFILE mapped to LOGDIR or ERROR_LOG (The directory to which the .log or .bad files are written)

More details about oracle External Table:

External tables are created using the SQL CREATE TABLE...ORGANIZATION EXTERNAL statement.

Steps to create external table:

1. Create directory where external data files are located.
2. Provide all required privileges
3. Create external table with CREATE TABLE...ORGANIZATION EXTERNAL statement.

More details about DB2 External table:

The below mentioned link has all the required details to create an external table on db2.

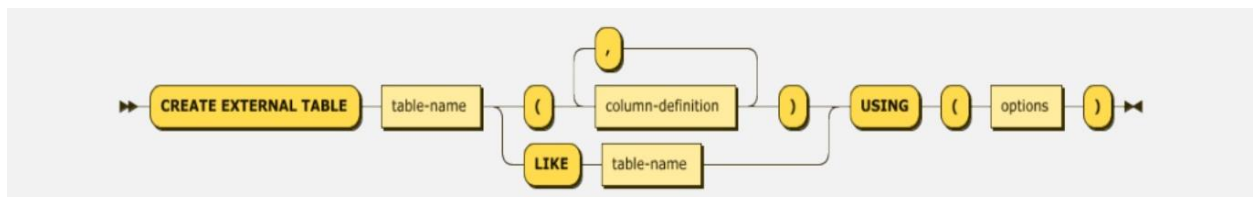
Link:

https://www.ibm.com/support/producthub/db2/docs/content/SSEPGG_11.5.0/com.ibm.db2.luw.sql.ref.doc/doc/r_create_ext_table.html

Please find below summarized information to get basic idea of Db2 external tables,

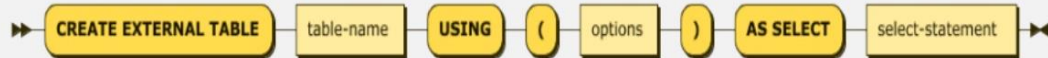
1. DB2 syntax for named external table:

The external table has a name and catalog entry similar to a normal table.



2. DB2 syntax for unloading data by using a transient external table:

The external table has a system-generated name of the form SYSTET<number> and does not have a catalog entry.



Supported datatypes in DB2 for external table in column definition,

- SMALLINT, INTEGER | INT, BIGINT
- DECIMAL | DEC | NUMERIC | NUM
- FLOAT(integer), REAL, DOUBLE, DOUBLE PRECISION, FLOAT, DECFLOAT
- CHARACTER | CHAR
- VARCHAR | CHARACTER VARYING | CHAR VARYING
- GRAPHIC, VARGRAPHIC
- NATIONAL CHARACTER | NATIONAL CHAR | NCHAR
- NATIONAL CHARACTER VARYING | NATIONAL CHAR VARYING | NCHAR VARYING | NVARCHAR
- NATIONAL CHARACTER LARGE OBJECT | NCHAR LARGE OBJECT | NCLOB
- BINARY, VARBINARY | BINARY VARYING
- CLOB | CHARACTER LARGE OBJECT | CHAR LARGE OBJECT
- DBCLOB
- BLOB | BINARY LARGE OBJECT
- DATE, TIME, TIMESTAMP(integer) or TIMESTAMP
- BOOLEAN