

Práctica 6

Carga de datos externos en Oracle. SQL*Loader

1. La herramienta SQL*Loader

Oracle dispone de una herramienta para llevar a cabo el movimiento de datos desde ficheros externos a tablas de una base de datos. Esta utilidad es el SQL*LOADER.

Sus características más importantes son:

- Carga de datos en una gran variedad de formatos
- Combina varios registros físicos en un sólo registro lógico
- Ejecución de filtros de carga
- Carga de varias tablas a la vez
- Carga de los datos desde una unidad de disco o cinta
- Generación automática de claves secuenciales únicas para una columna de una tabla.
- Informe detallado de la carga realizada, con mensajes acerca de los registros inválidos.

En la ejecución de SQL*LOADER intervienen los siguientes tipos de ficheros:

- **Fichero de Control:** Fichero que le indica a SQL*LOADER el formato de los datos.
- **Fichero de Datos:** Fichero con los datos que deben ser cargados en la(s) tabla(s) Oracle.
- **Fichero de Log:** Guarda las estadísticas de carga
- **Fichero de Error:** Almacena los registros incorrectos en caso de que los haya.
- **Fichero de Descartados:** Almacena los registros que no cumplen los criterios de selección o filtros.

En la figura 6.1 se muestra el tratamiento de los datos que lleva a cabo SQL*LOADER.

Dada la complejidad de la información almacenada en alguno de los ficheros, su contenido será descrito en una sección posterior.

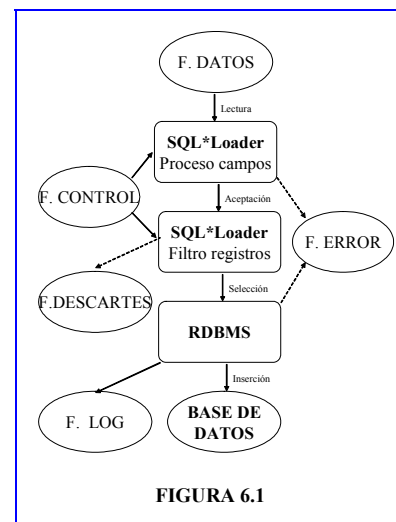


FIGURA 6.1

2. Carga de registros

2.1 Modos de carga

- **Carga Convencional.** En este tipo de carga los registros de datos son leídos y emplazados en un buffer temporal. Cuando dicho buffer se llena o ya no existen más datos que cargar, los registros lógicos se pasan a Oracle para la inserción en la tabla (mediante la sentencia INSERT). Este es el tipo de carga por defecto. Se debe usar cuando se tiene un número pequeño de tuplas a cargar.
- **Carga Directa** En este tipo de carga los registros se colocan en bloques de la base de datos formateados y son escritos directamente en la base de datos, saltándose algunos procesos del RDBMS. Esta carga es mucho más rápida que la carga *convencional*. Se debe usar la carga directa cuando se tiene una gran cantidad de datos para cargar en un corto periodo de tiempo. No en todos los casos es posible la carga *directa*. En Oracle 9, además, existe la **carga directa paralela** que permite que varias sesiones de carga de acceso directo carguen los mismos segmentos de datos de forma simultánea.

2.2 Requisitos de carga

Antes de proceder a la carga de registros en la base de datos se deben tener en cuenta las siguientes consideraciones:

- Las tablas que van a recibir los datos deben existir en la base de datos
- Se deben tener privilegios de *inserción* sobre la tabla o tablas en las que se van a cargar los registros.
- Se deben tener privilegios de *borrado* de la tabla a cargar si la carga se va a realizar con la opción REPLACE.

2.3 La utilidad sqlldr

Esta utilidad permite llevar a cabo las cargas de los registros en la base de datos. Su ejecutable está en el directorio *oracle/ora92/bin* y el formato de llamada es el siguiente:

```
sqlldr <opciones>
```

Donde opciones representa una serie de parámetros que se pueden especificar mediante posición o con el formato *nombre_de_parámetro=valor*.

Los parámetros válidos son:

- **userid**: usuario/contraseña@cadena de conexión
- **control**: Nombre del fichero de control
- **log**: Nombre del fichero de log
- **bad**: Nombre del fichero de errores
- **data**: Nombre del fichero de datos
- **discard**: Nombre del fichero de descartes.
- **discardmax**: Máximo número de descartes permitidos antes de abortar la ejecución. Por defecto: Todos
- **skip**: Número de registros lógicos a saltar desde el inicio del fichero de datos. Por defecto: 0
- **load**: Número de registros lógicos a cargar. Por defecto: All
- **errors**: Número de errores permitidos antes de finalizar la carga.
- **rows**: Número de tuplas dentro del buffer de carga en el modo convencional o entre inserciones en el modo directo. Por defecto 64 en el modo convencional y all en el directo.
- **bindsize**: Tamaño del buffer de carga. Por defecto 256000.
- **silent**: Suprime mensajes durante la ejecución.
- **direct**: Usar carga directa. Por defecto: False.
- **parallel**: Usar carga directa paralela. Por defecto: False.
- **parfile**: Fichero de parámetros opcional.
- **readsize**: Tamaño del buffer de lectura. Por defecto: 1048576.

Ejemplos:

```
sqlldr usuario/usuario@abd0 control=carga1.ctl log=carga1.log  
sqlldr userid=usuario/usuario@abd0 control=carga2.ctl direct=true
```

Nota:

Como ya hemos dicho antes, los parámetros de la línea de comandos se pueden especificar mediante posición o palabras clave.

Un ejemplo del primer caso es:

```
sqlldr usuario/usuario@abd0 control1.ctl
```

Un ejemplo del segundo es:

```
sqlldr control=control1.ctl userid=usuario/usuario@abd0
```

Se pueden especificar parámetros por posición antes, pero no después de los especificados por palabras clave.

Por ejemplo, está permitido:

```
sqlldr usuario/usuario@abd0 control=carga1.ctl logfile=carga1.log
```

pero no está permitido:

```
sqlldr usuario/usuario@abd0 control=carga1.ctl carga1.log
```

a pesar de que la posición del parámetro 'log' sea correcta.

3. El fichero de datos

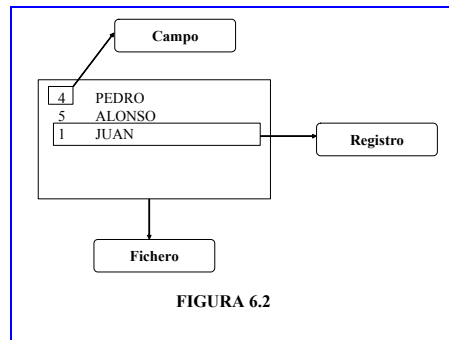
En este fichero se encuentran los datos que deben ser insertados en la base de datos. La estructura de un fichero de datos es la que se muestra en la figura 6.2.

3.1 Los ficheros

Los datos que deben ser cargados en las tablas o conjunto de tablas a través de SQL*Loader pueden encontrarse:

- Dentro del fichero de control. No es lo más habitual
- En uno o varios ficheros diferentes del de control.

El formato de los datos dentro del fichero puede ser muy variado. El fichero de datos debe encontrarse en formato carácter, no siendo posible la carga desde ficheros binarios, aunque provengan de la herramienta EXPORT de Oracle.



3.2 Los registros

Pueden venir formateados de dos formas:

- **Fijos:** Los datos del registro tienen todos la misma longitud, tipo y posición. Ejemplo:

```
4      PEDRO
5      ALONSO
6      MARIA
```

- **Variables:** Cada registro sólo tiene la longitud necesaria para contener los datos, utilizándose algún carácter como separador de campos. Los tipos de datos pueden variar de un registro a otro. Los sistemas operativos utilizan un carácter terminador (normalmente un salto de línea) para marcar el final de un registro. Ejemplo:

```
4;PEDRO
5;ALONSO
6;MARIA
```

3.3 Los campos

Pueden tener siempre la misma posición y longitud dentro de un registro o pueden variar si usan delimitadores.

4. El fichero de control

Le indica a SQL*Loader cómo interpretar el fichero de datos. Contiene información acerca de:

- Nombre de los ficheros de datos
- Formato de los datos
- Conjunto de caracteres
- Tipos de datos de cada campo
- Identificación de inicio y final de campo
- Qué tablas y columnas se deben cargar

Se pueden especificar los comandos de la línea de parámetros dentro del fichero de control usando la cláusula **OPTIONS**, que irá al principio del fichero de control:

```
OPTIONS (SKIP= n, LOAD=n...)
```

4.1 Datos de carga

Los datos que van a ser cargados en la base de datos pueden venir de dos formas:

- En uno o varios ficheros de datos independientes del fichero de control. Ejemplo:

```
LAOD DATA
INFILE clientes1.dat BADFILE clientes1.bad DISCARDFILE clientes1.dis
INFILE clientes2.dat
INFILE clientes3.dat DISCARDFILE clientes3.bad
INFILE clientes4.dat DISCARDMAX 10
INTO TABLE clientes
...
```

- Dentro del fichero de control con la cláusula BEGINDATA

```
LOAD DATA
INFILE *
INTO TABLE clientes
.....
BEGINDATA
1;JUAN
2;ROSA
3;CARLOS
```

4.2 Métodos de carga

Se puede especificar uno de los siguientes métodos de carga de datos en una tabla:

- **INSERT:** Retorna un error y termina la carga si la tabla contiene datos. Es la opción por defecto.
- **APPEND:** Añade nuevas tuplas a la tabla.
- **REPLACE:** Borra las tuplas en la tabla y carga los nuevos datos en su lugar.

Ejemplo:

```
LOAD DATA
INFILE *
APPEND
INTO TABLE clientes
...
```

Nota: La cláusula INTO TABLE indica la tabla donde se van a cargar los datos.

4.3 Unión de varios registros físicos para formar un registro lógico.

Puede ocurrir que un registro lógico se forme uniendo varios registros contiguos de datos.

Caso 1: Siempre se da la misma relación entre los registros físicos y lógicos.

En este caso utilizaremos la cláusula CONCATENATE para indicar cuántos registros físicos forman uno lógico.

Ejemplo:

```
LOAD DATA
INFILE *
APPEND
CONCATENATE 2
INTO TABLE clientes
.....
BEGINDATA
1;JUAN
JUAN XXIII, 2
3;CARLOS
```

ALFONSO VIII, 4

Caso 2: El número de registros físicos que forman uno lógico varía.

En este caso debemos utilizar la cláusula CONTINUEIF para determinar cuando ha finalizado un registro lógico.

El formato es CONTINUEIF THIS(X:Y) = 'cadena a comparar' donde X es la posición inicial e Y la posición final donde debe buscar la cadena a comparar. Si la encuentra, le estamos indicando que el registro lógico actual continúa en la línea siguiente. El registro lógico actual finalizará cuando incluyamos una línea que no comience por la cadena de caracteres especificada. A partir de la siguiente línea será otro registro lógico diferente.

En el ejemplo siguiente estamos indicando que si los 2 primeros caracteres de la fila son ** debe tomar también la siguiente.

Importante: las filas que no comiencen con la cadena de búsqueda deben dejar libres las posiciones (X:Y) pues no se tienen en cuenta para la inserción.

Ejemplo:

```
LOAD DATA
INFILE *
APPEND
CONTINUEIF THIS (1:2) = '**'
INTO TABLE clientes
.....
BEGIN DATA
**1; JUAN
      JUAN XXIII, 2
      2; ROSA
**3; CARLOS
      ALFONSO VIII, 4
```

4.4 Filtrado de registros a cargar

Se utiliza la cláusula WHEN para indicar si el registro lógico va a ser cargado en la tabla, o por el contrario se adjuntará al fichero de registros descartados.

Ejemplo:

```
LOAD DATA
INFILE *
APPEND
INTO TABLE clientes
WHEN (ID_CLIENTE <> 0) AND (ID_CLIENTE <> 5)
....
```

4.5 Delimitadores por defecto

Se utiliza cuando todos los campos terminan de una manera similar en el fichero de datos. El delimitador se indica con la cláusula FIELDS. Se puede indicar además con la cláusula ENCLOSED que algunos registros pueden tener campos que vayan delimitados además por un carácter como comillas simples, dobles, etc.

```
LOAD DATA
INFILE *
APPEND
INTO TABLE clientes
FIELDS TERMINATED BY ',' OPTIONALLY ENCLOSED BY '"'
...
```

4.6 Registros lógicos sin todos los campos

Dentro de un registro lógico puede ocurrir que uno o varios campos no se encuentren definidos. En este caso, se devolverá un error indicando que se ha llegado al final del registro sin tener valor para todos los campos de carga, excepto si se ha incluido la cláusula TRAILING NULLCOLS, en cuyo caso las columnas no incluidas se rellenarán a nulos.

4.7 Posiciones de los campos en el fichero

Cuando se cargan datos desde un fichero con registros fijos, hay que especificar el tamaño y la posición de cada uno de los campos en el registro. El primer carácter del registro lógico es el de la posición 1.

Ejemplo:

```
LOAD DATA
INFILE *
APPEND
INTO TABLE clientes
(id_cliente POSITION (01:03),
NOMBRE POSITION (04:22))
...
```

4.8 Cargas múltiples

Las cargas múltiples permiten:

- Cargar datos en diferentes tablas.
- Extraer múltiples registros lógicos desde un único registro de entrada.
- Distinguir diferentes formatos de entrada de registro.

Para llevar a cabo una carga de este tipo se deberán incluir dentro del fichero de control varias cláusulas INTO TABLE.

Ejemplo 1:

```
LOAD DATA
INFILE *
APPEND
INTO TABLE clientes
(id_cliente POSITION (01:03),
NOMBRE POSITION (04:22))
INTO TABLE clientes
(id_cliente POSITION (23:25),
NOMBRE POSITION (26:44))
BEGINDATA
1      JUAN                      2      ROSA
3      CARLOS                    4      PEDRO
```

Ejemplo 2:

```
LOAD DATA
INFILE *
APPEND
INTO TABLE clientes
WHEN recid = 1
(recid POSITION (1:1),
id_cliente POSITION (2:4),
nombre POSITION (5:23))
INTO TABLE proveedores
(recid POSITION (1:1),
id_proveedor POSITION (2:4),
nombre POSITION (5:23))
BEGINDATA
11     JUAN
21     JAVIER
22     PAULA
```

4.9 Generación de datos

Permiten introducir dentro de la carga datos no incluidos en los registros del fichero, que se generan automáticamente. Los datos que se pueden generar son:

- **RECNUM:** Contiene el número del registro lógico dentro del fichero de datos.

- **SYSDATE**: Contiene la fecha actual
- **CONSTANT**: Permite especificar un valor constante con el formato *CONSTANT valor*.
- **SEQUENCE**: Especifica un valor que depende de cómo definamos los parámetros de la secuencia. El formato de la secuencia es el siguiente:

```
SEQUENCE ({ n , MAX , COUNT} , incremento)
```

- **n**: indica que la secuencia comenzará en el número "n".
- **MAX**: la secuencia comenzará con el máximo valor del campo en la tabla más el incremento
- **COUNT**: la secuencia comenzará con el número de tuplas de la tabla más el incremento
- **incremento**: variación del valor de un registro al siguiente

Ejemplo:

```
LOAD DATA
INFILE *
APPEND
INTO TABLE clientes
(id_cliente RECNUM,
nombre POSITION (1:18))
...
```

4.10 Formatos de fechas.

Los campos que almacenan fechas pueden ser cargados en una gran multitud de formatos. Veamos un ejemplo de la definición de carga de un campo de tipo fecha:

```
...
INTO TABLE clientes
(id_cliente RECNUM,
 fecha_alta DATE(8) "DD/MM/YY")
BEGIN DATA
12/12/90
01/12/91
...
```

5. El fichero de log

Cuando SQL*Loader inicia la ejecución, crea un fichero de log. Si no puede ser creado, la ejecución finaliza. El fichero de log contiene información de la carga, incluyendo una descripción detallada de cualquier error que se haya producido.

6. El fichero de registros erróneos

Contiene registros rechazados en el proceso de carga. Las razones pueden ser alguna de las siguientes:

- Formato de registro inválido
- Intento de cargar un registro que produzca duplicidad de clave primaria dentro de la tabla.
- Paso de un valor nulo de un campo del registro a una columna con la restricción no nula ...

Los registros dentro del fichero de errores se encuentran en el mismo formato en que se encontraban dentro del fichero de carga. Este fichero puede ser utilizado como fichero de datos de una carga posterior una vez hayan sido corregidos los errores.

7. El fichero de registros descartados

Este fichero sólo se crea en caso de que vaya a contener datos. Contiene registros que no han pasado el filtro impuesto en el criterio de selección especificado en fichero de control. Son escritos en el mismo formato en que se encontraban dentro del fichero de datos.

8. Ejercicios

Pasos previos a la realización de los ejercicios:

- El tablespace DATOS_USUARIOX, el perfil ALUMNO_ABDX y el usuario USUARIOX1, con los privilegios que se le otorgaron en la práctica anterior, deben existir (ejercicios 1,2 y 3 de la práctica 5). Si no es así, créalos de nuevo.
- Ejecuta el script CreaObjetosLoader para USUARIOX1, para crear las 2 tablas que se detallan a continuación.
- Comprueba que las tablas se han creado correctamente.

TABLA DE CATEGORIAS

CAMPO	TIPO	LONGITUD
Id_categoria	Numérico	4
Nombre	Carácter	20

TABLA DE PROVEEDORES

CAMPO	TIPO	LONGITUD
Id_proveedor	Numérico	4
Nombre	Carácter	10
Direccion	Carácter	30
Telefono	Carácter	9
Fecha_nacimiento	Fecha	

Ejercicio 1

Cargar dentro de la tabla *categorías* los siguientes registros:

```
1; CATEGORIA A
2; CATEGORIA B
3; "CATEGORIA C"
```

- Dichos registros deberán ser incluidos dentro del fichero de control.
- Interpretar la salida obtenida en el fichero de log.
- Realizar una consulta a la tabla para comprobar que se ha cargado correctamente.

Ejercicio 2

Cargar dentro de la tabla *categorías* los registros que se encuentran en el fichero "ej2ldr.dat".

- Añadir a las tuplas ya existentes en la tabla
- Interpretar la salida obtenida en el fichero de log.
- En caso de que exista algún registro erróneo, modificar el fichero de errores y utilizarlo como fichero de datos en una nueva carga.
- Al final comprobar que la tabla *categorías* contiene los registros cargados.

Ejercicio 3

Cargar dentro de la tabla *categorías* los registros que se encuentran en el fichero "ej3ldr.dat".

- Rellenar la columna 'Id_Categoría' con un número secuencial que parte del máximo valor actual que tiene la columna dentro de la tabla.
- Añadir a las tuplas ya existentes en la tabla
- Interpretar la salida obtenida en el fichero de log.
- Consultar la información almacenada en la tabla *categorías*.

Ejercicio 4

Cargar dentro de la tabla *proveedores* los registros que se encuentran en el fichero "ej4ldr.dat" y cuyo contenido es:

```
$JUAN;ALFONSO IX, 23;  
225577;09-06-1936  
PEDRO;JUAN XXIII, 15;;15-05-1968  
$PEPE;;  
221212  
ANA;; 212525;15-03-19545;  
ROSA;;;12-12-1970
```

- Dichos registros deberán ser incluidos dentro del fichero de control.
- Mientras que se encuentre un '\$' en la primera posición de un registro físico, el siguiente registro físico formará parte del mismo registro lógico.
- No deben ser cargados aquellos proveedores que no tienen teléfono asociado.
- Se deben borrar todos los datos existentes con anterioridad en la tabla.
- El campo Id_proveedor se rellenará para cada registro lógico, con el valor siguiente al máximo valor existente en la tabla de proveedores en ese momento.
- Interpretar la salida obtenida en el fichero de log.
- Interpretar el contenido de la tabla *proveedores* después de realizar la carga.