



DOCUMENTACIÓN ULTRA DBSCRIPT

Septiembre 2018

Versión 1.0

CONTROL DE CAMBIOS

<i>Fecha</i>	<i>Versión</i>	<i>Autor</i>	<i>Revisado por</i>
01/10/2018	1.0	Guillermo Vallespir Wood	Guillermo Vallespir Wood

1. INTRODUCCIÓN

Ultra DBScript es la evolución natural de su antecesor DBScript. Ultra DBScript es un estándar XML para scripts de definición y manipulación de datos en bases de datos SQL.

Ultra DBScript permite especificar los siguientes tipos de acciones:

- Conectarse a bases de datos
- Crear tablas, secuencias, claves e índices
- Modificación de estructura de tablas (agregar, editar y eliminar columnas)
- Llenado de tablas con datos predefinidos
- Cargar tablas con datos provenientes de archivos de texto delimitados o con columna de ancho fijo.
- Cargar tablas provenientes de otras bases de datos
- Ejecutar sentencias SQL.

Esta versión de Ultra DBScript sólo es compatible con los siguientes motores de bases de datos:

- MySQL

Para entender el presente documento se deben tener nociones de XML.

La aplicación “Ultra_DBScript.jar” permite ejecutar este tipo de scripts XML.

1.1. Notas generales del uso de XML

Atributos en cursiva

Los atributos que se encuentren en cursiva son aquellos utilizados en las versiones de DBScript y que actualmente se encuentran obsoletos en esta versión de Ultra DBScript, pero debido a asuntos de compatibilidad se han mantenido dentro de ella, sin embargo, podrían ya no ser compatibles en versiones posteriores, además que por razones de simplificación de conceptos algunos han sido reemplazados por otros con designaciones más significativas en versiones posteriores de esta.

Caracteres especiales

Los siguientes caracteres deben ser sustituidos por las secuencias de escape indicadas cuando se les desee usar dentro del texto de un tag o como valor de un atributo.

<i>Carácter</i>	<i>Secuencia de escape</i>
&	&
<	<
>	>
'	'
“	&qout;

Así, por ejemplo, `<SQL> select fld from tbl where fld >= 4 </SQL>` debería escribirse como:

```
<SQL> select fld from tbl where fld &gt;= 4 </SQL>
```

Para evitar hacer estos cambios en los caracteres se puede encerrar el texto entre los tags **<![CDATA[** y **]]>**, que indican al intérprete XML que todo lo que contiene es texto literal, que incluso puede contener símbolos especiales como '`<`' u otros caracteres especiales (puntuación, acentos, símbolos, etc)

```
<SQL><![CDATA[select fld from tbl where fld >= 4]]></SQL>
```

Comentarios

Todo el texto que se encuentre entre los caracteres “`<!--`” y “`-->`” se considera como comentario y Ultra DBScript no ejecutará todas las líneas comentadas.

```
<!-- Proceso de carga -->
```

Comodines SQL

Se utilizan dentro de sentencias SQL para reemplazar a funciones particulares y específicas de todas las bases de datos soportadas, cuya denominación es dependiente del motor de base de datos que se esté utilizando.

Tags que soportan esta funcionalidad: SQL, SQL2TABLE, SQL2TXT.

Se listan los comodines disponibles y la función a la que reemplazan para cada base de datos en particular:

<i>Comodín</i>	<i>SQL Server</i>	<i>MySQL</i>	<i>Oracle</i>	<i>ODBC</i>
%ifnull	ISNULL	IFNULL	NVL	NO APLICA
%curdate	GETDATE()	CURDATE()	SYSDATE	NO APLICA
%now	GETDATE()	NOW()	SYSDATE	NO APLICA
%isnull	IS NULL	IS NULL	--	--
%isnotnull	--	IS NOT NULL	--	--

Ejemplo:

```
<SQL>
select sc.codigo_sku, sc.codigo_centro, floor(%ifnull(sc.stock_fisico, 0) -
%ifnull(sc.backorder, 0) from requerimiento_sku rs, compania c where
rs.codigo_centro_roigen=sc.codigo_centro, fecha=%curdate
</SQL>
```

Tipos de Datos Genéricos

Ultra DBScript proporciona una serie de tipos de datos genéricos para la construcción y modificación de tablas en bases de datos. Ultra DBScript interpreta estos tipos de datos genéricos y los traduce a su equivalente dependiendo del DBMS que se esté utilizando. A continuación se listan todos los tipos de datos genéricos y sus equivalencias:

<i>Tipo genérico</i>	<i>MySQL</i>	<i>SQL Server</i>	<i>Oracle</i>	<i>Postgres</i>
int / integer	INT	INT	NUMBER	
unsigned int / uint	INT UNSIGNED	INT	NUMBER	
bigint	BIGINT	BIGINT	BIGINT	
unsigned bigint / ubigint	BIGINT UNSIGNED	BIGINT	BIGINT	
mediumint	MEDIUMINT	INT		
Unsigned mediumint / umediumint	MEDIUMINT UNSIGNED	INT		
smallint	SMALLINT	INT		
unsigned smallint / usmallint	SMALLINT UNSIGNED	INT		
tinyint	TINYINT	INT	NUMBER	
unsigned tinyint / utinyint	TINYINT UNSIGNED	INT		
double	DOUBLE	FLOAT		
float	FLOAT	FLOAT	NUMBER	
numeric	NUMERIC			
decimal	DECIMAL	DECIMAL		
real	REAL			
boolean	BIT			
varchar / string	VARCHAR	NVARCHAR	VARCHAR2	
char	CHAR	CHAR	CHAR	
text	TEXT	NVARCHAR	CLOB	
longtext	LONGTEXT	NVARCHAR		
mediumtext	MEDIUMTEXT	NVARCHAR		
smalltext	SMALLTEXT			
date	DATE	DATE		
datetime	DATETIME	DATETIME	DATE	

year	YEAR	SMALLINT	NUMBER
time	TIME	TIME2	
blob	BLOB		
smallblob	SMALLBLOB		
mediumblob	MEDIUMBLOB		
longblob	LONGBLOB		
set	SET		
enum	ENUM		

1.2 Estructura Ultra DBScript

Encabezado del archivo XML

Todos los archivos XML debe comenzar con el siguiente encabezado que lo identifica como documento XML.

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
```

Inicio y Fin de un Script

Un archivo XML puede contener uno o más scripts, cada uno de los cuales se inicia con el tag **<DBSCRIPT>** y debe finalizar con el tag **</DBSCRIPT>**. Dentro de este tag se definen todas las sentencias y acciones a realizar. Un documento XML puede contener una o más tag DBSCRIPT.

Los atributos de DBSCRIPT son los siguientes

<i>Atributo</i>	<i>Descripción</i>
id	Identificador del script
author	Autor del script. Nombre del desarrollador
date	Fecha de creación del script.
desc	Descripción del proceso realizado por el script.

Ejemplo:

```
<DBSCRIPT id="Test Stockdown" author="Guillermo Vallespir W." date="20180928"
desc="Carga de datos Stockdown">
.
.
.
.
.
</DBSCRIPT>
```

Atributos generales

Por lo general las etiquetas XML (todas) usadas para definir procesos o tareas tienen los siguientes atributos en común:

Atributo	Descripción
id	Identificador del elemento de proceso a realizar. Este id también es usado para ser referenciado desde otras ubicaciones del script ejecutado. (esta acción es soportada por los tags FILE , DATABASE , FILE_LIST , PARAM , CONST , etc)
fails	Indica si la tarea o acción está en modo “A prueba de fallos” (fails=0) o sensible a errores (fails=1, predeterminado). Útil si se requiere que algún proceso sea opcional. Si fails=0, y ocurre algún error de proceso, se notifica del error, se guarda en el log y continúa la ejecución del script sin detenerse.
loglevel	Define el modo del log o registro. Cuando loglevel es 0 (loglevel=0) no se registrará ningún mensaje durante el procesamiento del script que define este tag (excepto mensajes de error).
enabled	Activa (enabled=1) o desactiva (enabled=0) el procesamiento del tag. Por defecto: enabled=1. Si el valor es distinto a 0 (o el valor está vacío) se tratará como activa (enabled=1).
desc	Descripción del tag o proceso. Debe ser una descripción corta y es opcional.

1.3. Parametrización de arranque

Ultra DBScript permite la realización de parametrizaciones y configuraciones al momento del arranque de Ultra DBScript a través de argumentos pasados por línea de comandos que alterarán la ejecución normal de Ultra DBScript. A continuación se muestran los argumentos posibles pasados por argumentos a través de línea de comandos:

Usage: [--startX] [--non-verbose] [(-c|--config) config1;config2;...;configN] [(-C|--config-file) <file>] [(-m|--mode) <mode>] [(-?|--help) <help>] [-h] file1 file2 ... fileN

Argumento	Descripción
--startX	Arranca Ultra DBScript en modo visual. Si se activa este parámetro se omitirán todo el resto de los parámetros.
--non-verbose	Arranca Ultra DBScript en modo non-verbose, las salidas por pantalla serán reducidas.
-c --config <config1;config2,...,configN>	Establece una serie de configuraciones al momento del arranque de DBScript. Para más información sobre los atributos de configuración soportados, revisa el tag <CONFIG>. Ejemplo: -c “dbscript.terminal.ignore_colors=yes;dbscript.terminal.ignore_pauses=yes”

-C --config-file <config-file>	Establece un archivo .txt o .ini con la configuración que será cargada y procesada por Ultra DBScript al momento del arranque. Si se establece este argumento, el argumento -c o --config no será procesado. Para más información sobre los atributos de configuración soportados, revisa el tag <CONFIG>. Ejemplo --config-file /home/stockdown5/config.ini
-m --mode <mode>	Establece el modo de arranque de Ultra DBScript. Ultra DBScript tiene la capacidad de iniciar en modo <i>dbscript</i> , <i>neodomain</i> o <i>branches</i> , ejecutando las acciones propias para cada modo de ejecución. Ejemplo: --mode neodomain. <i>Por defecto mode = dbscript.</i>
-? --help <tag>	Muestra la ayuda con respecto a un tag determinado. Si se establece este campo, todos los demás campos serán omitidos y Ultra DBScript sólo mostrará la ayuda por pantalla del tag especificado. Ejemplo: -? include.

2. <INCLUDE> Inclusión de documentos XML

El tag <INCLUDE> permite incluir documentos XML adicionales al original, desviando el curso de la ejecución de los tags al nuevo documento, para después continuar con la ejecución original en caso que hubiesen quedado tags pendientes. El tag <INCLUDE> puede ser ejecutado en cualquier sección del documento XML original. Esta versión soporta un número ilimitado de tags include en el documento, además de soportar tags recursivos, esto permite definir tags include dentro de archivos XML llamados por otros tag include.

Atributo	Descripción
xml_file	El archivo xml a cargar. Procesa el documento y continúa con la ejecución.

Ejemplos:

```
<INCLUDE xml_file="import_tecnored.xml" />
<INCLUDE xml_file="schema_sd4.xml" />
```

3. <PARAM> Creación de parámetros globales del script

El tag <PARAM> permite crear un parámetro a ser utilizado dentro de los **atributos** en los tags del documento XML. Esto permite definir una variable global (el parámetro) que puede ser referenciado desde cualquier atributo (o contenido de texto de un tag) reemplazando la expresión por el valor entregado. Si desea usar parámetros para **consultas SQL** (en tags <SQL>) puede definirlos usando el

tag especializado <PARAM_SQL>.

Atributo	Descripción
id	Es el nombre del parámetro que se reemplazará or el valor indicado en el atributo value.
token	Representa el carácter delimitador del parámetro en las expresiones que lo usarán (Ej: %PARAMETRO%). Por defecto se utiliza '%'. <i>Este campo es opcional.</i>
value	Valor por el que será reemplazado el parámetro en las expreciones encerradas por el <i>token</i> . Si está activado <i>sqlvalue</i> y la consulta no obtuvo valores, se utiliza este valor por defecto.
sqlvalue	Especifica una consulta SQL de donde obtendrá el valor. Si se especifica esta consulta, se asignará a value el primer valor obtenido en el primer campo del resultado de la consulta. (Ej: “select nombre_pais from pais where codigo='CL'” ==> “Chile”). <i>Este campo es opcional.</i>
db	Identificador de la base de datos asociada a la consulta en sqlvalue. Debe haberse definido previamente con el tag <u><DATABASE></u> cuyo id sea este valor. <i>Este campo es obligatorio sólo si se define sqlvalue. Si no se define sqlvalue, este campo no se considerará.</i>
const	Define el parámetro como constante, es decir, su valor una vez asignado no podrá ser cambiado durante la ejeccuión de Ultra DBScript. <i>Este campo es opcional y su valor por defecto es 0 (const=0)</i>

Además existen algunos parámetros ya definidos, relacionados a fechas, horas y variables del sistema.

%YYYY%	: El año actual en cuatro dígitos (2018)
%YY%	: El año actual en dos dígitos (18)
%MM%	: El mes actual en dos dígitos (01-12)
%DD%	: El día actual en dos dígitos (01-31)
%hh%	: La hora actual en dos dígitos (00-24)
%mm%	: Minutos actuales en dos dígitos (00-59)
%ss%	: Segundos actuales en dos dígitos (00-59)
%YYYYMMDD%	: Corresponde a la fecha actual en formato YYYYMMDD (Ej: 20181001)
%hhmmss%	: Corresponde a la hora actual en formato hhmmss (Ej: 105923)
%DATE%	: Alias de %YYYYMMDD%
%TIME%	: Alis de %hhmmss%
%USER%	: Nombre del usuario que está ejecutando Ultra DBScript
%VERSION%	: Versión de Ultra DBScript que se está ejecutando
%DWEK%	: Nombre del día actual de la semana (lunes – domingo)
%AUTHOR%	: Autor del archivo XML que actualmente se está procesando, definido en etiqueta author del DBScript.
%XML_PROCESS%	: Nombre y ruta del archivo XML que actualmente se está

procesando.

Ejemplo 1 (Definir un parámetro, usarlo para definir otro parámetro, o como un atributo de otro tag).

```
<PARAM id="pais" value="Chile" token="%" />
value = Chile
<PARAM id="nombre_archivo" value="pedido_%pais%" token="%" />
value = pedido_Chile
<PARAM id="mifecha" value="%YYYY%-%MM%-%DD%" token="%" />
value = 2018-10-01
<FILE id="archivo_exportacion" name="%nombre_archivo%_%mifecha%.txt" />
name = pedido_Chile_2018-10-01.txt
```

Ejemplo 2 (Definir un parámetro desde una consulta SQL y usarlo para generar un nombre de archivo).

```
<PARAM id="numpedido" value="" token="%" db="STOCKDOWN" sqlvalue="select
MAX(num_pedido) from pedido" />
<FILE id="archivo_exportacion" name="pedido_%numpedido%.txt"></FILE>
```

4. <CONST> Creación de constantes globales del Script

El tag **<CONST>** es una extensión del tag [<PARAM>](#), con la diferencia que en vez de crear un parámetro con valor modificable durante la ejecución del Script, crea un parámetro con valor que nunca podrá ser cambiado o modificado después de asignado. La constante podrá ser utilizada dentro de los **atributos** en los tags del documento XML. Esto permite definir una constante global (la constante) que puede ser referenciada en cualquier atributo (o contenido de texto de un tag) reemplazando la expresión por el mismo valor entregado siempre. Si desea usar constantes para las **consultas SQL** (en tags **<SQL>**) puede definirlos usando el tag [<PARAM_SQL>](#) y asignarles el valor **const=1**.

Atributo	Descripción
id	Es el nombre de la constante que se reemplazará por el valor indicado en el atributo value.
token	Representa el carácter delimitador del parámetro en las expresiones que lo usarán. (Ej: %CONSTANTE%). Por defecto se utiliza '%'. <i>Este campo es opcional.</i>
value	Valor por el que será reemplazado la constante en las expresiones encerradas por el token. Si está activado sqlvalue y la consulta no obtuvo valores, se utiliza este valor por defecto.
sqlvalue/source	Especifica una consulta SQL de donde obtendrá el valor. Si se especifica esta consulta, se asignará a value el primer valor obtenido en el primer campo del resultado. (Ej: "select nombre_pais from pais where codigo='CL'" ==> "Chile"). *Recurda que una vez asignado un valor, este no podrá ser cambiado.
db	Identificador de la base de datos asociada a la consulta en sqlvalue. Debe haberse definido previamente un tag <DATABASE> cuyo id sea este valor. <i>Este campo es opcional y si no se define el atributo</i>

sqlvalue.

5. <PARAM_SQL> Creación de parámetros para sentencias SQL

El tag <PARAM_SQL> permite crear un parámetro a ser utilizado dentro de sentencias SQL en el documento XML. Esto permite definir una variable global (el parámetro) que puede ser utilizada muchas veces, logrando evitar hacer cambios en muchos lugares cuando debe cambiarse el valor de este atributo y este tiene muchas ocurrencias dentro del documento. Tags que soportan esta funcionalidad: [SQL](#), [SQL2TABLE](#), [SQL2TXT](#).

Atributo	Descripción
id	Representa el parámetro utilizado dentro del documento en una o más ocurrencias dentro de los tags SQL, SQL2TABLE y SQL2TXT. Este parámetro debe ser utilizado dentro del documento XML.
value	Este es el valor por el que será reemplazado el parámetro 'id' dentro del documento.
token	Representa el carácter que a va a ser utilizado para encerrar (se utiliza uno al principio y otro al final de la expresión) los parámetros dentro de las sentencias SQL en el documento. Por defecto se utiliza la barra vertical: ' '. <i>Este campo es opcional.</i>
const	Establece si el parámetro SQL es una constante o no. Si se establece el campo const=1, significa que el parámetro será una constante y su valor no podrá ser cambiado una vez haya sido asignado. Por defecto const = 0. <i>Este campo es opcional.</i>

Ejemplos:

```
<PARAM_SQL id="tabla_test" value="tmp_test_01" token="$" />
<PARAM_SQL id="tabla_sku_centro" value="tmp_sku_centro_test_xml"
token="%"></PARAM_SQL>
```

6. <PATH> Creación de alias para directorios

El tag <PATH> permite asociar a un alias el nombre y ruta de un directorio, para ser utilizados de forma genérica por la aplicación en el paso de parámetros. El nombre y ruta del archivo debe definirse en la parte de **texto** del tag XML. Referenciado en tags: [TXT2TABLE](#), [MOVE_FILE](#), [SQL2TXT](#)...

Atributo	Descripción
Id	Representa el alias utilizado dentro del documento en una o más ocurrencias dentro del documento XML para un directorio en especial.

El nombre y ruta del directorio debe usarse en parte de texto del tag XML. Este puede ser tanto relativo como absoluto. Ejemplo:

```
<PATH id="archivo_test">C:\Documents and Settings\Administrador</PATH>
```

7. <FILE> Creación de alias para archivos

El tag <FILE> permite asociar a un alias el nombre y ruta de un archivo, para ser utilizado de forma genérica por la aplicación en el paso de parámetros. El nombre y ruta del archivo debe definirse en la parte de **texto** del tag XML. Referenciado en tags: TXT2TABLE, [MOVE_FILE](#), SQL2TXT, ...

Atributo	Descripción
id	Representa el alias utilizado por este archivo para ser referenciado por otros tags.
ftp	Opcional. El ID de un tag <FTP> definido con información de acceso a un FTP.
ftpfile	Opcional. Define la ruta del archivo FTP a descargar. El archivo se descargará en la ruta y nombre definido en el texto del tag FILE, sobrescribiendo el archivo actual, si existiera. * Si no agrega el atributo <i>ftp</i> , se necesita especificar el atributo <i>ftphost</i> (Host/IP del servidor FTP). Si el acceso no es anónimo (usuario <i>anonymous</i>), también debe especificar <i>ftpuser</i> (usuario FTP) y <i>ftppass</i> (contraseña FTP).
ftphost	Se no está definido el atributo <i>ftp</i> se debe especificar el actual. Éste representa el host/IP de la conexión FTP.
ftpuser	Si no está definido el atributo <i>ftp</i> se debe definir el actual. Éste representa el nombre de usuario de acceso al FTP, si es que el acceso no es anónimo.
ftppassword/ftppass	Si no está definido el atributo <i>ftp</i> se debe definir el actual. Éste representa la contraseña del usuario de acceso FTP, si es que el acceso no es anónimo.
read_interval	Tiempo de espera en segundos para chequear lectura de archivos si ha cambiado de tamaño (por defecto 5). Si pasado este tiempo el archivo ha cambiado de tamaño, vuelve a chequear una vez más hasta que el archivo no cambie.
retries	Número de intentos para obtener el archivo. Por defecto, solo intenta una vez.
interval	Si se establece el número de intentos, este parámetro indica el intervalo en segundos para los intentos siguientes. Si se alcanza el número de intentos sin éxito, el proceso generará un error.
onerror	Permite definir un script o programa a ejecutar si el intento actual falla. Es útil para efectuar alguna acción alternativa para obtener el archivo desde otra ubicación, por ejemplo. Se ejecuta entre cada nuevo intento de chequear el archivo. Véase el tag <EXEC>.
onsuccess	Permite definir un script o programa a ejecutar si el intento actual termina exitosamente.
check	Semejante a <i>onerror</i> , permite ejecutar un script o programa luego de encontrar el archivo. Si el programa devuelve un error, el proceso generará un error. Esto es útil si se requiere efectuar un chequeo de contenido o de formato del archivo especificado, o efectuar correcciones automáticas en al archivo para su posterior procesamiento. Véase el tag <EXEC>

Ejemplos: (definición simple de un archivo, y definición descargando desde un servidor FTP previamente definido):

```
<FILE id="archivo_sku">C:\test\tmp_sku_centro_test_xml.txt</FILE>
<FILE id="external" ftp="server1" ftpfile="/download/ext.csv">C:\download\ext.csv
</FILE>
```

8. <FILE_LIST> Creación de alias para una lista de archivos

El tag <FILE_LIST> permite asociar a un alias una lista de archivos (con nombre y ruta, usando comodines), para ser utilizados de forma genérica por la aplicación. El nombre y ruta de la lista de archivos debe definirse en el texto del tag. La ruta se define entre los tags. Referenciado en tags TXT2TABLE, [MOVE_FILE](#).

Atributo	Descripción
id	Representa el identificador utilizado como referencia dentro del documento XML.
optional	Indica si el listado es opcional o se requiere que existan los archivos.
retries	Número de intentos para obtener el archivo. Por defecto, sólo intenta una vez.
interval	Si se establece el número de intentos, éste parámetro indica el intervalo en segundos para los intentos siguientes. Si se alcanza el número de intentos sin éxito, el proceso generará un error.
onerror	Permite definir un script o programa a ejecutar si el intento actual falla. Es útil para efectuar alguna acción alternativa para obtener el archivo desde otra ubicación, por ejemplo. Se ejecuta entre cada nuevo intento de chequear el archivo. Véase el tag <EXEC>
check	Similar a <i>onerror</i> , permite ejecutar un script o programa luego de encontrar el archivo. Si el programa devuelve un error, el proceso generará un error. Esto es útil si se quiere efectuar un chequeo de contenido o de formato del archivo especificado, o ejecutar correcciones automáticas en el archivo para su posterior procesamiento. Véase el tag <EXEC>
read_interval	Tiempo de espera en segundos para chequear lectura del archivo si ha cambiado de tamaño (por defecto 5). Si pasado este tiempo el archivo ha cambiado de tamaño, vuelve a chequear una vez más hasta que el archivo no cambie de tamaño.
append	Es un valor 1 ó 0 que indica si se van a agregar archivos con errores a la lista actual. Si el valor es 1 se agregan a la lista actual.
db	Identificador de la base de datos. Debe haberse definido previamente la base de datos mediante el tag <DATABASE>, a cuyo id se hace referencia.
chequear_lista/check_list	Expresión para buscar chequeos de archivos en las tablas listado_archivo y listado_archivo_detalle de la Base de Datos.
sqlcheck	Es una consulta SQL la que retorna 1 ó 0, si es 1 el proceso se detiene y en caso contrario este continuará normalmente. Ejemplo: (SELECT

if(hour(curdate()) > 12, 1, 0) as detener), o sea, si la hora actual es mayor a 12 se retornará 1 y como consecuencia el proceso se detiene.

Ejemplo: (grupo de archivos de texto que comienzan por "ls_test_" en la carpeta c:\test)

```
<FILE_LIST id="lista_test">C:\test\ls_test_*.txt</FILE_LIST>
```

9. <MOVE_FILE> Mover archivos a una ruta definida

El tag <MOVE_FILE> permite mover uno o varios archivos a una carpeta determinada.

Atributo	Descripción
id	Representa el identificador utilizado durante la ejecución.
file	Nombre del archivo a cargar, especificando su ruta. La ruta default es la ubicación del archivo XML que contiene el script. Se puede utilizar dentro del nombre del archivo un asterisco (*) como wildcard, para poder utilizar múltiples archivos con nombres similares. También puede utilizarse un ID que identifique el contenido de un tag <FILE>, con referencia a otro archivo.
file_list	Debe usarse un id que especifique al contenido de un tag <FILE_LIST>, quien hace referencia a una lista de archivos. Si utiliza 'file_list', el atributo 'file' no será considerado.
path_dest	Es la ruta de destino a la que se moverán los archivos.
optional	Indica si la carga del archivo es obligatoria o no para la ejecución del script. En caso de ser '0' (obligatorio), la ejecución se detendrá si no encuentra el archivo. En caso de ser '1' (opcional), el script seguirá ejecutándose aunque no encuentre el archivo. Valor por defecto '0'. <i>Este campo es opcional</i>
overwrite	Indica si sobrescribirá el archivo si es encontrado en el directorio de destino. Si el atributo es '1' (overwrite), el archivo será sobrescrito. En caso de ser '0' (skip), terminará sin efectos la ejecución del tag, el archivo será ignorado y el script seguirá ejecutándose normalente. El valor por defecto es '1' (overwrite). <i>Este campo es opcional.</i>

Ejemplos:

```
<MOVE_FILE id="movercsv" file_list="test_csv" path_dest="C:\destino_test\" />
<MOVE_FILE id="moveflv" file_list="test_flv" path_dest="C:\destino_test\"
optional="1" overwrite="0" />
```

10. <DATABASE> Conexión a una base de datos

El tag <DATABASE> permite efectuar una conexión a una base de datos. Sus atributos son:

Atributo	Descripción
id	Identificador de la base de datos. Debe ser un identificador único para cada base de datos, pues se usará para referenciar a la base de datos desde otros tags (como SQL, SQL2TABLE).
type / db_type	Tipo de base de datos. Los valores soportados por esta versión de Ultra DBScript son: mysql, oracle, mssql, odbc, postgresql, sysbase, mariadb, mongodb.
host	Nombre del host/IP de la base de datos. Suele ser una dirección IP, aunque en el caso de motores de bases de datos como Oracle, corresponde al nombre de la BD en el TNSNAMES. En el caso de ODBC se puede dejar en blanco.
port	Este atributo especifica el puerto de conexión a la base de datos. <i>Este campo es opcional y su valor por defecto variará según el tipo de base de datos, por ejemplo: para conexiones Oracle su valor por defecto es 1523 y para bases de datos MySQL su valor por defecto es 3306.</i>
database	Nombre de la base de datos (Para Oracle también puede ser una definición de TNS completa. Véase http://www.orafaq.com/wiki/Tnsnames.ora)
user	Nombre de usuario de la base de datos.
password	Contraseña de acceso a la base de datos.
jtdsPipes	Este atributo indica sólo para bases de datos MS SQL Server si utiliza una conexión TCP/IP o Named Pipes (canalizaciones con nombre). El valor por defecto es '0': utilizar conexiones vía TCP/IP. En caso de necesitar las conexiones por Pipes, definir este atributo en '1'. Debe tener en cuenta que esta opción sólo funciona bajo sistemas operativos Windows. <i>Este campo es opcional: la selección del tipo de conexión SQL Server es determinada automáticamente.</i>
url	Alternativamente se puede prescindir del uso de los campos <i>host</i> , <i>port</i> , <i>database</i> , <i>user</i> y <i>password</i> agrupándolos todos en el campo <i>url</i> siguiendo el siguiente patrón: <code>jdbc:tipo://host:port/database?parametro=valor&parametro2=valor2</code> donde <i>valor</i> y <i>valor2</i> son los valores asignados a <i>parametro</i> y <i>parametro2</i> respectivamente, además el encabezado <code>jdbc:tipo</code> depende del tipo de base de datos que deseemos conectar. La conveniencia de utilizar esta forma de conexión es que podemos entregar parámetros extras adicionales según lo requiramos. <i>Este campo es opcional.</i>
class	Este atributo solo se usa en conjunto con el atributo <i>url</i> , e indica la clase a implementar para la conexión, sus valores posibles son: com.mysql.jdbc.Driver (mysql), oracle.jdbc.driver.OracleDriver (oracle), net.sourceforge.jtds.jdbc.Driver (mssql, jtds), com.microsoft.jdbc.sqlserver.SQLServerDriver (sql2000),

sun.jdbc.odbc.JdbcOdbcDriver (odbc), com.ibm.db2.jcc.DB2Driver (DB2), org.postgresql.Driver (postgresql). *Este campo es obligatorio si se está utilizando el campo url, en caso contrario es opcional.*

Ejemplos:

```
<DATABASE id="STOCKDOWN" type="mysql" host="burns" database="sd_trial" user="sd"
password="sd" />
<DATABASE id="ora10" db_type="oracle" host="ora" database="orcl" user="sd"
password="sd" port="1523" />
<DATABASE id="psql_url" db_type="postgresql"
url="jdbc:postgresql://psql:1523/psql_db?user=sd&password=sd"
class="org.postgresql.Driver" />
```

Se pueden incluir tantos tags <DATABASE> como sean necesarios.

Es obligatorio incluir los tags <DATABASE> antes de los tags que manipulan las bases de datos, pero no es necesario que todos sean declarados al comienzo del documento, pero sí antes de ser utilizados.

11. <FTP> Conexión a un servidor FTP

El tag <FTP> permite asociar un alias a un servidor FTP junto a la información de acceso a éste. Tags que soportan esta funcionalidad: [FILE](#).

Atributo	Descripción
id	Representa el alias utilizado para hacer referencia a esta conexión FTP en otros tags que lo definan.
host	Indica el nombre del host o IP del servidor FTP.
user	Optional. Nombre del usuario FTP para acceder al servidor FTP (por defecto: <i>anonymous</i>).
password	Optional. Clave del usuario FTP para acceder al servidor FTP (por defecto: <i>anonymous</i>).
port	Optional. Define el número de puerto TCP/IP que utiliza el servidor FTP para conectarse. (por defecto: 21)
proxyhost	Es el Host/IP del servidor proxy de conexión.
proxyport	Es el puerto del servidor proxy de la conexión.

Ejemplo:

```
<FTP id="ftpserver1" host="192.168.101.200" user="usrstk01" password="passwstk"
port="2100"></FTP>
```

12. <CREATE_TABLE> Creación de una tabla en base de datos

El tag <CREATE_TABLE> permite crear una table en una base de datos. La tabla no debe existir previamente o el script no tendrá efecto. Esta funcionalidad está disponible para todos los tipos de bases de datos soportados. Los atributos de este tag son:

Atributo	Descripción
db	Identificador de la base de datos en la que se creará la tabla. Debe haberse definido previamente la base de datos a través del uso del tag <DATABASE> , a cuyo <i>id</i> se hace referencia.
name	Nombre de la tabla que será creada
collate	Se especifica el juego de caracteres a utilizar para las cadenas de texto.
tablespace	Para bases de datos Oracle especifica en cuál TABLESPACE será creada la tabla. Si no se especifica será creada en el Tablespace por defecto del usuario.
extends	Permite definir una tabla basada en los campos ya existentes de otra tabla. Se debe especificar el nombre de la tabla de la cual se extiende.
before	Especifica qué operación se efectuará antes del proceso de creación de la tabla. Ejemplo: before="drop", elimina la tabla antes de crearla.

Dentro de este tag deben especificarse los campos de la tabla mediante el tag <FIELDS>. Dentro del tag <FIELDS> se identifica cada campo en particular con tags <FIELD>, cuyos atributos son los siguientes:

Atributo	Descripción
name	Nombre del campo en la tabla
type / fld_type	Los tipos son especificados de modo genérico, Ultra DBScript se encarga de traducirlos al tipo particular del DBMS que corresponda, para más información vea la tabla de tipos genéricos de datos de Ultra DBScript más abajo.
length	Largo del campo del tipo de dato. No todos los tipos de datos aplican el atributo length. <i>Este campo es opcional. Su valor por defecto dependerá del tipo de dato especificado en el atributo type.</i>
nullable	Indica si el campo acepta nulos. ('1' = NULL, '0' = NOT NULL). <i>Por defecto '1' (NULL).</i>
default	Valor por omisión para el campo. <i>Este atributo es opcional.</i>
primary_key	Se especifica si el campo es una clave primaria, sus valores posibles son '0' (No es clave primaria) o '1' (es clave primaria). <i>Este campo es opcional. Por defecto: '0' (No es clave primaria).</i>
autoinc / autoincrement	Define si el campo es autoincremental. Su incremento está definido en 1 unidad. Este atributo no tiene efecto sobre bases de datos de tipo Oracle. Sus valores pueden ser: '0' (no autoincremental), o '1' (autoincremental). <i>Este campo es opcional. Por defecto: '0' (no autoincremental).</i>

Ejemplo:

```
<CREATE_TABLE db="STOCKDOWN" name="upl_leadtime_csv">
  <FIELDS>
    <FIELD name="codigo_sku" type="varchar" length="30" nullable="0"
default="" />
    <FIELD name="media_leadtime" type="float" nullable="1" default="" />
    <FIELD name="desvstd_leadtime" type="float" nullable="1" default="" />
    <FIELD name="codigo_proveedor" type="int" nullable="0" autoinc="1" />
  </FIELDS>
</CREATE_TABLE>
```

13. <ALTER_TABLE> Modificación de una tabla

El tag <ALTER_TABLE> permite modificar la estructura de una tabla ya existente en la base de datos. Actualmente se puede **agrega**, **modificar** y **eliminar** campos en una tabla. Si se requiere agregar un campo ya existente, producirá un error, al igual que si se desea modificar o eliminar un campo no existente, producirá un error. Los atributos de este tag son semejantes a [<CREATE_TABLE>](#)

Atributo	Descripción
db	Identificador de la base de datos en la que se creará, modificará o eliminarán campos de la tabla. Debe haberse definido previamente la base de datos a través del tag <DATABASE>, a cuyo <i>id</i> se hace referencia.
name	Nombre de la tabla ya existente en la base de datos.
collate	Se especifica el juego de caracteres a utilizar para cadenas de texto. <i>Este campo es opcional.</i>

Dentro de este tag deben especificarse las etiquetas para agregar campos <ADD_FIELDS>, modificar campos <MODIFY_FIELDS> o eliminar campos <DROP_FIELDS>. Se pueden incluir cuántas etiquetas add_fields, modify_fields y drop_fields se requieran dentro de un mismo tag <ALTER_TABLE>. La ejecución de estas etiquetas será recursivamente, sin embargo, primero procesa las etiquetas ADD_FIELDS en orden, luego las etiquetas MODIFY_FIELDS en orden y por último las etiquetas DROP_FIELDS en orden.

13.1. <ADD_FIELDS>

El tag <ADD_FIELDS> agrega nuevos campos (columnas) a la base de datos, se define dentro del tag ADD_FIELDS cada nuevo campo con el tag <FIELD>. Sus atributos son los siguientes:

Atributo	Descripción
name	Nombre del campo que será creado en la tabla
type / fld_type	Los tipos son especificados de modo genérico, y traducidos al tipo particular del DBMS que corresponda. Para más información, ver tabla de tipos genéricos Ultra DBScript
Length	Largo del campo del tipo de dato. No todos los tipos de datos requieren el atributo length. Si se especifica el atributo length a un tipo de dato que no lo requiere (Ej: longtext) éste no será considerado. <i>Este campo es</i>

	<i>opcional. Su valor por defecto dependerá del tipo de dato especificado en el atributo type.</i>
Nullable	Indica si el campo acepta nulos. Observación: