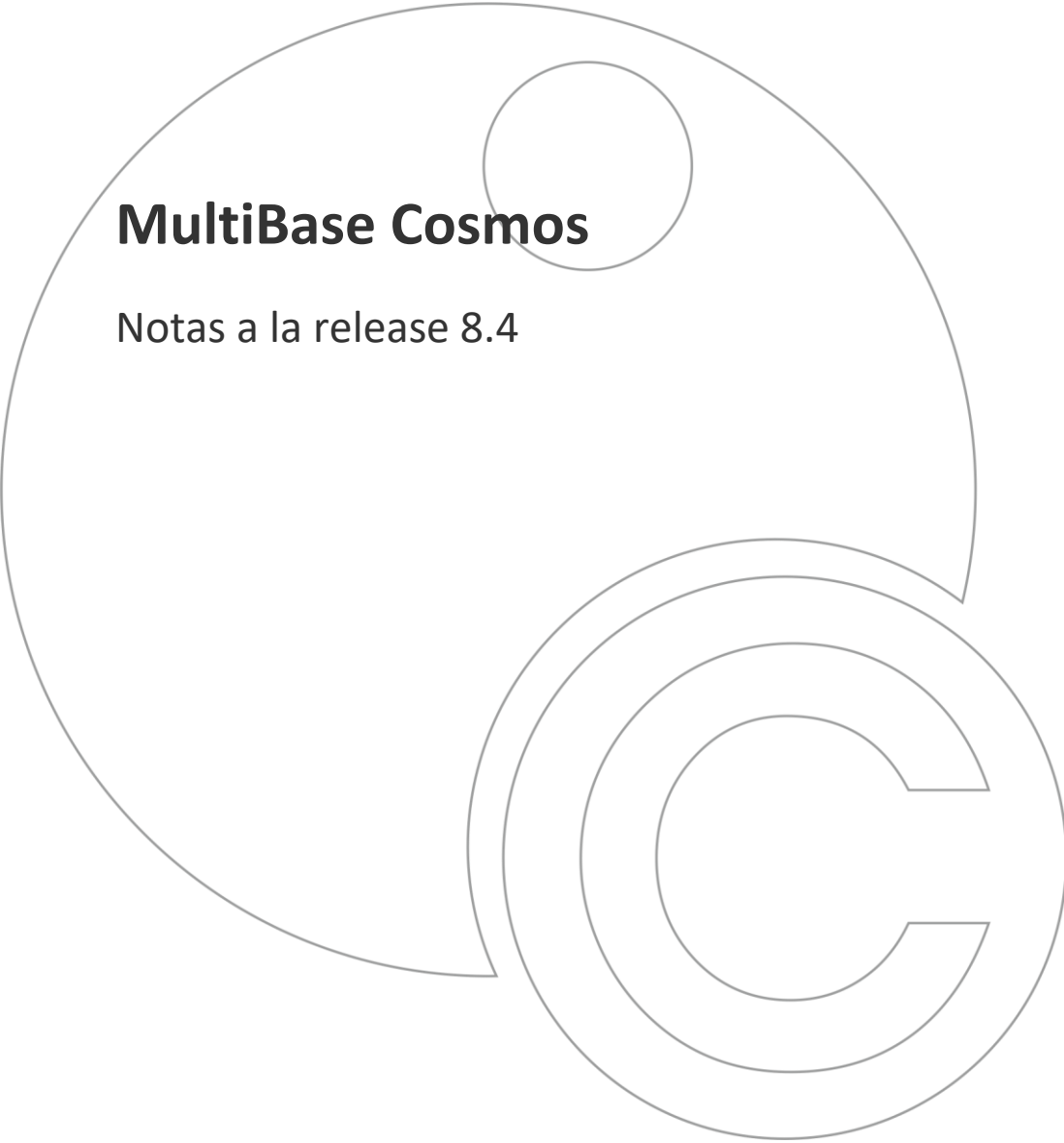




MultiBase Cosmos

Notas a la release 8.4

BASE100

BASE 100, S.A.
www.base100.com

Índice

1. IMPLEMENTACIONES	3
1.1 RUNTIME	3
1.2 COSMOS WEBSERVER	3
1.3 ENTORNO DE DESARROLLO	4
1.3.1 <i>Aceleradores de teclado</i>	4
1.3.2 <i>Compilador</i>	4
1.3.3 <i>Depurador</i>	4
1.3.4 <i>Editor de constantes</i>	4
1.4 REGISTRO DE COSMOS	4
2. MEJORAS	5
2.1 CLASE FORMTABLE	5
2.2 CLASE STREAM.....	5
2.3 CLASE JSON	5
2.4 CLASE MODULE.....	5
2.5 COMPILADOR.....	5
2.6 ENTORNO DE DESARROLLO	5
2.7 PROCEDIMIENTOS ALMACENADOS	5
3. MÉTODOS	6
3.1 CLASE CHAR	6
3.2 CLASE SQLCURSOR	6
3.3 CLASE SQLSERVER	7
4. VARIABLES DE ENTORNO	8
5. CORRECCIONES	9
5.1 RUNTIME	9
5.1.1 <i>Visualización</i>	9
5.1.2 <i>Exportación Excel</i>	9
5.1.3 <i>Clase Json</i>	9
5.1.4 <i>Cliente ODBC</i>	9
5.1.5 <i>Clase Smallint</i>	9
5.1.6 <i>Clase FormTable</i>	9
5.2 ENTORNO DE DESARROLLO	9
5.3 COMPILADOR.....	9
5.4 COSMOS WEBSERVER	10
5.5 CTSQL	10

1. Implementaciones

1.1 Runtime

- Los métodos «FtpPutFileEx» y «urlSendComand» ahora soportan los protocolos FTPS (ftp implícito) y FTP con TLS (explícito).
- Extracción de la información de la base de datos en una cadena de caracteres en formato «JSON». Para ello se ha implementado o ampliado la funcionalidad de los siguientes métodos:
 - Método «SelectTupleJson» de la clase «SqlServer».
 - Método «FetchTupleJson» de la clase «SqlCursor».
 - Método Unload de la clase SqlServer.
 - Método SelectWindow de la clase SqlServer. Botón «Export».
 - CSQL.
 - Salida de una consulta al pulsar el botón «Export».
 - Instrucción «unload».

En los métodos Unload, SelectWindow, en la instrucción unload y en salida de una consulta desde «CSQL», para que la descarga se realice en formato «JSON», la extensión del fichero deberá ser «.json».

Estos ficheros de texto contienen todas las filas de las tablas descargadas en formato «JSON», y su estructura es un array de «JSON». Cada elemento del array corresponde a un «JSON» con la información de cada fila de la tabla. El nombre de propiedad es el nombre de la columna y su valor es el valor de la columna en esa fila.

- Nuevo método CompareStrict de clase Char que permite comparar dos cadenas de caracteres byte a byte.
- Control List Box. Se amplía el tamaño de las cabeceras y del scroll automáticamente cuando se amplía el tamaño de la fuente.
Para que este proceso de lleve a cabo hay que definir la variable de entorno «ADJUSTLISTHEADERHEIGHTBYFONT» con valor «TRUE» o «YES».

1.2 Cosmos WebServer

- Posibilidad asignar un fichero de log por cada servicio definido en Cosmos WebServer. En versiones anteriores a la versión 8.4 todas las entradas se añadían en el mismo fichero («Cwslog.log»).
- Posibilidad de poder definir la ubicación y el nombre del fichero («Cwslog.log») que registra los errores, «warning» y salidas del método «Trace» del «runtime» de Cosmos.

En ambos casos habrá que definir la variable de entorno «CWSLOGFILE».

1.3 Entorno de desarrollo

1.3.1 Aceleradores de teclado

Habilitar/deshabilitar el code insight y el deep search. La combinación de teclas se indica a continuación:

- «CTRL+Alt+I» para habilitar/deshabilitar code insight.
- «CTRL+Alt+D» para habilitar/deshabilitar code deep search.

1.3.2 Compilador

Nueva opción en el entorno de desarrollo que permite que, al compilar un proyecto completo, no deje de mostrar mensajes cuando se llegue a los 100 errores, como sucedía en versiones anteriores. Esta opción se podrá habilitar en el entorno de desarrollo. Está deshabilitada por defecto.

1.3.3 Depurador

Posibilidad de desacoplar el panel de variables «watch» del depurador y convertirlo en una ventana flotante.

1.3.4 Editor de constantes

Posibilidad de poder arrastrar para duplicar constantes dentro del mismo módulo. La combinación de teclas que deberá utilizar será «CTRL+clic», arrastra y soltar.

Para realizar la acción debe seleccionar el ítem con las teclas indicadas. Al soltar el ratón se mostrará una ventana para introducir el nombre de la nueva constante.

1.4 Registro de Cosmos

Mostrará un mensaje de error que describa la razón por la cual no se ha podido llevar a cabo el proceso.

2. Mejoras

2.1 Clase Formtable

Se ha llevado a cabo una optimización en los objetos de tipo FormTable con columnas lookup. No se realizará la consulta a la tabla referenciada si en el valor de la columna lookup es nulo. Esta implementación afecta a la tabla de cabecera como a las de líneas.

Para que las consultas no se lleven a cabo habrá que definir la variable de entorno «OPTIMIZELOOKUPNULLVALUES» con valor «1».

2.2 Clase Stream

Se ha implementado la posibilidad de acceso a rutas superiores de 127 caracteres.

2.3 Clase Json

Se ha optimizado el proceso de adición de elementos a un array «JSON».

2.4 Clase Module

- Método «MsgText». A partir de la versión 8.4 de Cosmos admite, en el segundo parámetro del método, una ruta absoluta a una unidad compartida de red. En versiones anteriores solo admitía una unidad red si ésta estaba mapeada.
- Método «GetUrlFileEx» admite la sintaxis «ftpes://» para FTP Seguro Explícito. En versiones anteriores admitía la descarga de ficheros desde servidores con FTP Seguro Explícito, pero no admitía la sintaxis estándar «ftpes://».

2.5 Compilador

Al compilar todo el proyecto, si no encuentra un fichero de iconos, mostraba un *message box* cada vez que encontraba un error de fichero de iconos no encontrado. A partir de la versión 8.4 de Cosmos, el mensaje de error se mostrará en la ventana de salida de errores.

2.6 Entorno de desarrollo

Se ha ampliado el tamaño del campo «variable» en los siguientes controles:

- La pestaña «Storage» del control «Edit field» de la clase Form.
- La pestaña «Storage» del control «Variable» de la clase Page.

2.7 Procedimientos almacenados

Se ha optimizado el proceso de ejecución de procedimientos almacenados.

3. Métodos

3.1 Clase Char

CompareStrict

Permite comparar dos cadenas de caracteres byte a byte.

Sintaxis:

```
CompareStrict (compareStr as Char) return integer
```

Parámetros:

compareStr	Cadena de caracteres con la que se va a comparar el objeto Char.
------------	--

Retorna:

Un entero. Los valores son:

0	Si las dos cadenas de caracteres son iguales.
1	Si el objeto char es mayor que la cadena de caracteres que recibe como parámetro el método.
-1	Si el objeto char es menor que la cadena de caracteres que recibe como parámetro el método.

Este método no tendrá en cuenta el idioma instalado en el PC como sucede cuando se comparan dos cadenas de caracteres con los operadores ==, < y >.

3.2 Clase SqlCursor

FetchTupleJson

Este método lee una tupla de la tabla derivada devuelta por un cursor abierto previamente con el método «Open» de la misma clase y retorna una cadena de caracteres con formato «JSON».

Sintaxis:

```
FetchTupleJson (oem as Boolean default FALSE) return Char
```

Parámetros:

Oem	Si es TRUE carga el fichero con el conjunto de Caracteres OEM y si es FALSE lo carga en ANSI.
-----	---

Retorna:

Devuelve la tupla en un Char en formato «JSON».

3.3 Clase SqlServer

SelectTupleJson

Este método permite ejecutar una instrucción SELECT recogiendo la información de la primera fila encontrada en una cadena de caracteres con formato «JSON».

Sintaxis:

```
SelectTupleJson (select as Char, oem as Boolean default FALSE) return Char
```

Parámetros:

Select	Instrucción select.
Oem	Si es TRUE carga el fichero con el conjunto de Caracteres OEM y si es FALSE lo carga en ANSI.

Retorna:

Un Char en formato «JSON» con el nombre de las columnas y los valores correspondientes.

4. VARIABLES DE ENTORNO

ADJUSTLISTHEADERHEIGHTBYFONT

Esta variable de entorno permite ajustar el tamaño de la cabecera en los controles tipo «List Box» al tamaño de la fuente.

Se deberá definir con valor TRUE o YES en la sección «Environment» del fichero de configuración del proyecto o de Cosmos. Si la altura resultante del cálculo es menor que la altura original de la cabecera de la lista, el tamaño de la cabecera será el original, como si no se hubiera definido la variable.

CWSLOGFILE

Permite definir la ruta del fichero «cwslog.log». Se define en el fichero de configuración de Cosmos WebServer sección «[Server]».

LONGVARCHARASCHAR

Esta variable de entorno permite que los tipos de dato «Long Varchar» en una conexión ODBC se traten como «Char».

Esta variable debe definirse en la conexión ODBC.

NORESTOREBUTTONICONS

En el caso de que un control «Box» tenga asociado un «FormTable» y dentro del control «Box» esté definido un control «Button», si se cambia el icono del botón en tiempo de ejecución con el método «SetIcon», y posteriormente se ejecuta cualquier comando de tabla («New», «Add», etc), automáticamente se restaura su icono original. En el caso de que no deseemos que este sea el comportamiento por defecto, será necesario definir esta variable de entorno con valor «YES» o «TRUE».

Esta variable se debe definir en la sección «Environment» del fichero de configuración del Cosmos o del proyecto.

OPTIMIZELOOKUPNULLVALUES

Si en un FormTable existe una columna de lookup, Cosmos obtiene su valor realizando una consulta a la tabla referenciada, aunque el valor de la columna sea «null».

Si se define esta variable con valor igual «1» y el valor de la columna definida en un lookup es «null», no realizará la consulta a la tabla referenciada. Tanto en la cabecera como en las líneas.

Esta variable se debe definir en la sección «Environment» del fichero de configuración del Cosmos o del proyecto, con el método «SetOption» de la clase Module o Form.

5. Correcciones

5.1 Runtime

5.1.1 Visualización

El repintado de las pantallas era lento al definir en la plantilla del fichero «CUSTOMCONTROLS» se definen los controles «Box» con bordes. Se ha optimizado el proceso de repintado.

Si se modifica el icono de un control «Button» que pertenece a un «FormTable», al ejecutar cualquier método de «FormTable», como «New», restaura el icono original y no respeta la modificación. Para cambiar este comportamiento, definir la variable de entorno «NOESTOREBUTTONICONS» con valor TRUE.

5.1.2 Exportación Excel

Al exportar el contenido de un control «Grid» a fichero con extensión «xlsx» no se respeta el tipo de la fuente.

5.1.3 Clase Json

No funcionaba correctamente la modificación de elementos de un array de «JSON».

5.1.4 Cliente ODBC

Al realizar una consulta en la que incluía un literal «select 'nombre cliente' cliente from tabla» el dato no se muestra correctamente. Mostraba el carácter «?».

Para que el dato se muestre correctamente hay que definir en la conexión ODBC la variable «LONGVARCHASCHAR» con valor TRUE.

5.1.5 Clase Smallint

La llamada al método «WeekDayName3» provocaba un error en ejecución y se aborta la ejecución.

5.1.6 Clase FormTable

Al definir la variable de entorno «LOOKUPOUTERJOIN» con valor «TRUE» se ejecutaban algunas select de más para consultar el valor de los lookup.

5.2 Entorno de desarrollo

Cuando se utilizan los aceleradores «Ctrl+flecha Izq» y «Ctrl+ Flecha dcha», si se encuentran con un carácter "¿", se queda parado.

5.3 Compilador

Se producía un error de protección general al compilar un módulo en el que se concatenaba a una cadena de caracteres el carácter «%» sin haber indicado el carácter de concatenación «+».

5.4 Cosmos WebServer

La función «CosHttpGetReturnCode» no retornaba el código HTTP cuando el verbo que se envía en la petición era «DELETE». En este caso, siempre retornaba 0 cuando debería devolver 200 o 404.

5.5 CTSQL

Al ejecutar la función «AVG» el gestor de base de datos retornaba el siguiente mensaje «división por cero». Este error se produce desde la versión 7.8.3 de Cosmos.