



LibreOffice
The Document Foundation

Guía de primeros pasos

Capítulo 13

Primeros pasos con Macros

Utilizar la grabadora de macros

Derechos de autor

Este documento tiene derechos de autor © 2011–2012 por sus colaboradores, tal como aparecen en la lista en la sección titulada **Colaboradores**. Puede distribuirse o modificarse bajo los términos de la *GNU General Public License*, versión 3 o posterior, o la *Creative Commons Attribution License*, versión 3.0 o posterior.

Todas las marcas registradas que aparecen dentro de esta guía pertenecen a sus dueños legítimos.

Colaboradores

Juan C. Sanz

Jorge A. Guzmán Soriano

Comentarios y sugerencias

Por favor, dirija cualquier clase de comentario o sugerencia acerca de este documento a: localizacion@ooo.es.org

Agradecimientos

Este capítulo está basado en el capítulo 13 de la Guía de comienzo de OpenOffice.org. Los colaboradores de este capítulo son:

Andrew Pitonyak

Jean Hollis Weber

Juan C. Sanz

Fecha de publicación y versión del programa

Versión en español publicada el 7 de mayo de 2012.

Basada en la versión 3.3 de LibreOffice

Nota para usuarios de Mac

Algunas pulsaciones de teclado y opciones de menú son diferentes en un Mac de las usadas en Windows y Linux. La siguiente tabla muestra algunas sustituciones comunes para las instrucciones dadas en este capítulo. Para una lista detallada vea la ayuda de la aplicación.

<i>Windows/Linux</i>	<i>Equivalente en Mac</i>	<i>Efecto</i>
Herramientas → Opciones opción de menú	LibreOffice → Preferencias	Acceso a las opciones de configuración
Clic con el botón derecho	Control+clic	Abre menú contextual
Ctrl (Control)	⌘ (<i>Comando</i>)	Utilizado con otras teclas
F5	Shift+⌘+F5	Abre el navegador
F11	⌘+T	Abre la ventana de estilos y formato

Contenido

Derechos de autor.....	2
Nota para usuarios de Mac.....	2
Su primera macro.....	4
Crear una macro simple.....	4
Ejecutar la macro.....	5
Ver y editar la macro.....	6
Los comentarios comienzan por REM.....	7
Definir subrutinas con SUB.....	7
Definir variables usando DIM.....	7
Poniendo toda la macro junta.....	8
Crear una macro.....	8
Un ejemplo complicado.....	9
Ejecutar la macro rápidamente.....	12
A veces la grabadora de macros falla.....	12
El dispatch framework.....	12
Cómo usa la grabadora de macros el dispatch framework	12
Otras opciones.....	13
Organización de macros	13
¿Dónde se almacenan las macros?.....	16
Importar macros.....	16
Descargar macros para importar.....	18
Cómo ejecutar una macro.....	18
Barra de herramientas.....	20
Elemento de menú.....	20
Atajos de teclado.....	20
Evento.....	21
Extensiones.....	22
Escribir macros sin la grabadora.....	23
Encontrar más información.....	23
Material incluido.....	24
Recursos online.....	24
Material impreso y eBooks.....	24

Su primera macro

Una macro¹ es una secuencia de comandos o pulsaciones de tecla, guardados para un uso posterior. Un ejemplo de macro simple es aquella que “escribe” su dirección. El lenguaje de macros de LibreOffice es muy flexible, permitiendo la automatización tanto de tareas simples como de tareas complejas. Las macros son especialmente útiles para repetir la misma tarea una y otra vez.

Las macros de LibreOffice se escriben normalmente en un lenguaje llamado StarBasic, o abreviando Basic. Aunque puede aprender Basic y escribir macros, la curva de aprendizaje para crear macros desde cero es muy empinada. El método usual de aprendizaje para un principiante es usar la grabadora de macros incluida, la cual graba sus pulsaciones y las guarda para un uso posterior.

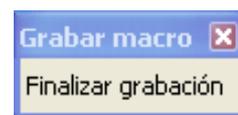
La mayoría de las tareas en LibreOffice se llevan a cabo “enviando comandos” los cuales son interceptados y usados. La grabadora de macros trabaja grabando los comandos que se envían (vea “El dispatch framework” en la página 12)

Crear una macro simple

Imagínese introduciendo repetidamente una información simple. Aunque puede almacenar la información en el portapapeles, si utiliza el portapapeles para algo más, el contenido cambia. Almacenar el contenido como una macro es una solución simple. (En algunos casos, incluyendo el ejemplo usado aquí, es mejor solución utilizar AutoTexto).

- 1) Utilice **Herramientas** → **Macros** → **Grabar macro** para empezar a grabar una macro.

Se muestra una pequeña ventana para que sepa que LibreOffice está grabando.



- 2) Escriba la información deseada o realice una serie adecuada de operaciones. En este caso yo he escrito mi nombre, *Andrew Pitonyak*.
- 3) Haga clic en el botón **Finalizar grabación** para terminar la grabación, guardar la macro y mostrar el diálogo Macros Basic de LibreOffice (vea Figura 1).
- 4) Asegúrese de abrir el contenedor de bibliotecas llamado *Mis macros*. Busque la biblioteca llamada *Standard* en Mis macros. Tenga cuidado, *cada* contenedor de bibliotecas tiene una biblioteca llamada *Standard*. Seleccione la biblioteca *Standard* y haga clic en **Nuevo módulo** para crear un módulo nuevo para contener a la macro.

¹ Macro es una apócope de macroinstrucción, por eso se usa en femenino

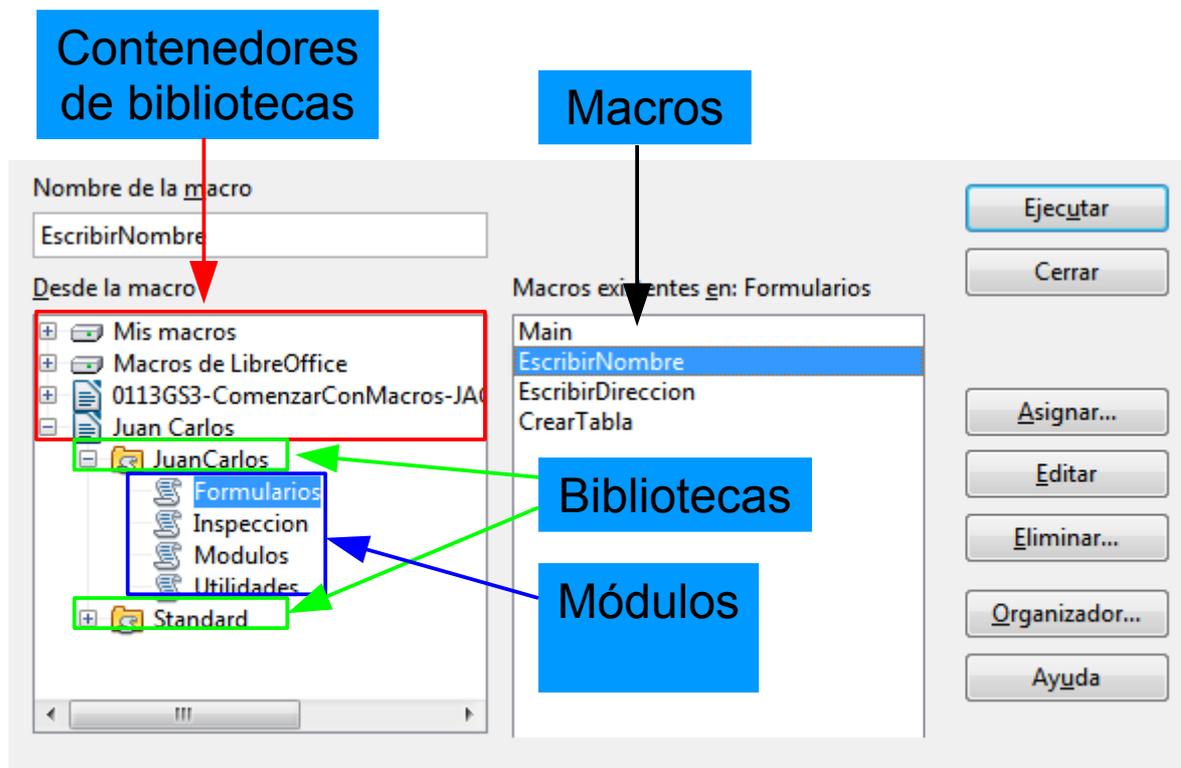


Figura 1: Dialogo de organización de macros

- 5) El nombre por defecto del nuevo módulo es Module1; elija un nombre más adecuado. Aunque aún no es muy descriptivo, yo he usado *Grabado*. Escriba un nombre descriptivo y pulse **Aceptar** para crear el módulo. Se muestra el diálogo Macros Basic de LibreOffice de nuevo mostrando el módulo nuevo.



Figura 2: Dar al módulo un nombre descriptivo

- 6) Resalte el módulo recién creado. En la esquina superior izquierda escriba el nombre de la macro, como "EscribirMiNombre" y haga clic en **Guardar** para guardar la macro.

Si ha seguido todos los pasos, la biblioteca Standard ahora contendrá un módulo llamado Grabado que contiene la macro EscribirMiNombre, como se muestra en la Figura 3. Cuando LibreOffice crea un módulo nuevo, crea automáticamente la macro llamada Main, como puede ver en la Figura 3.

Ejecutar la macro

Seleccione **Herramientas** → **Macros** → **Ejecutar macro** para abrir el diálogo Selector de macro (ver Figura 3). Seleccione la macro recién creada y haga clic en **Ejecutar**.

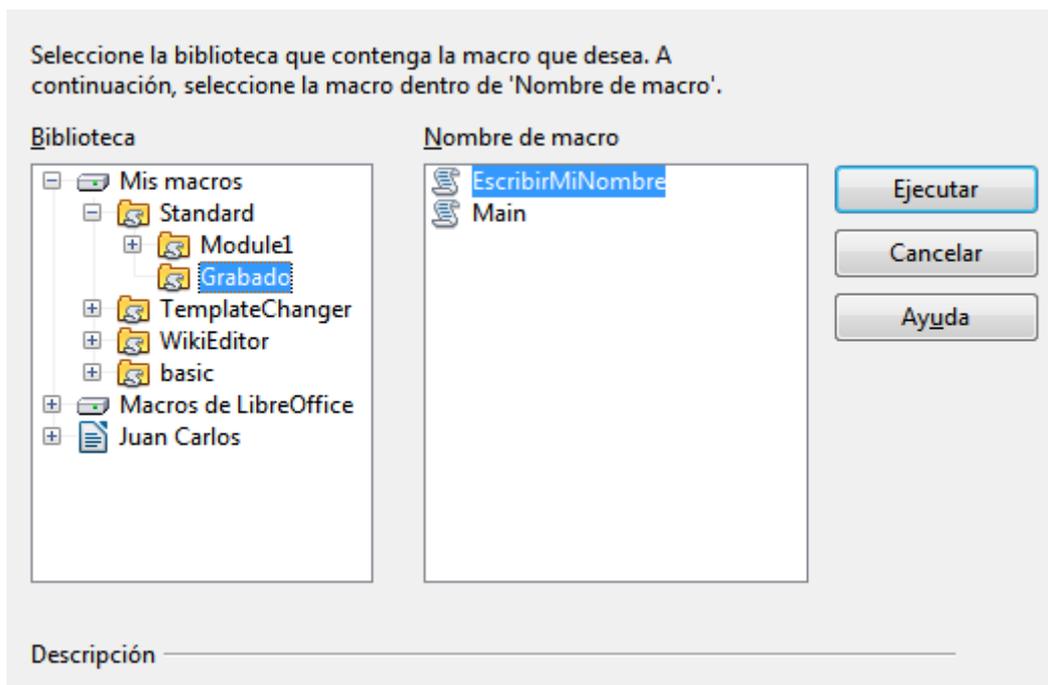


Figura 3: Seleccione su macro y haga clic en Ejecutar

Hay otros métodos de ejecutar una macro. Por ejemplo, use **Herramientas** → **Macros** → **Organizar macros** → **LibreOffice Basic** para abrir el organizador de macros, el cual tiene también un botón **Ejecutar**. El autor, ávido escritor de macros, prefiere el organizador de macros porque el diálogo generalmente se abre más rápido, aunque la selección de la macro puede ser ligeramente más lenta.

Ver y editar la macro

Puede ver y editar la macro que acaba de crear. Utilice **Herramientas** → **Macros** → **Organizar macros** → **LibreOffice Basic** para abrir el diálogo Macros de LibreOffice (vea Figura 3). Seleccione la nueva macro y haga clic en **Editar**

Listado 1: Macro generada "EscribirMiNombre".

```
REM ***** BASIC *****
Sub Main

End Sub

sub EscribirMiNombre
rem -----
rem define variables
dim document as object
dim dispatcher as object
rem -----
rem get access to the document
document = ThisComponent.CurrentController.Frame
dispatcher = createUnoService("com.sun.star.frame.DispatchHelper")

rem -----
dim args1(0) as new com.sun.star.beans.PropertyValue
```

```

args1(0).Name = "Text"
args1(0).Value = "Andrew Pitonyak"

dispatcher.executeDispatch(document, ".uno:InsertText", "", 0,
args1())
end sub

```

La macro del Listado 1 no es tan complicada como parece al principio. Aprender unas pocas cosas ayuda significativamente a entender las macros generadas. La discusión comienza con características en la parte superior del listado y su descripción. Si quiere evitar detalles, entonces simplemente cambie el texto “Andrew Pitonyak” por el que desee insertar en la posición actual del cursor.

Los comentarios comienzan por REM

La palabra clave REM, abreviatura de *remark*, comienza un comentario de macro. Todo el texto después de REM (en la misma línea) se ignora. Se puede usar también una comilla simple para comenzar un comentario.

Sugerencia

StarBasic no es sensible a las mayúsculas, por lo que tanto REM como Rem o rem sirven para comenzar un comentario. Si usa constantes simbólicas definidas por la API, es más seguro asumir que los nombres son sensibles a las mayúsculas —las constantes simbólicas son un tema avanzados y generalmente no son necesarias para la gente que usa la grabadora de macros.

Definir subrutinas con SUB

Las macros individuales se almacenan en subrutinas definidas con la palabra clave SUB. El final de una subrutinas se indica por las palabras END SUB. El código comienza definiendo la subrutina llamada Main, que está vacía y no hace nada. La siguiente subrutina EscribirMiNombre contiene el código generado.

Sugerencia

LibreOffice crea una subrutina vacía llamada Main cuando crea un módulo.

Hay temas avanzados que van más allá del alcance de este documento, pero conocerlos podría ser interesante:

- Se puede escribir una macro de modo que se puedan pasar valores a la subrutina. Los valores se llaman argumentos. Las macros grabadas no aceptan argumentos.
- Otra clase de subrutina se llama función. Una función es una subrutina que devuelve un valor. Para definir una función se utiliza la palabra clave FUNCTION en lugar de SUB. Las macros generadas son siempre del tipo SUB.

Definir variables usando DIM

Se puede escribir información en un trozo de papel de modo que se pueda leer después. Una variable, como un trozo de papel, contiene información que se puede leer y cambiar. La declaración DIM es similar a reservar un trozo de papel para almacenar un mensaje o nota.

La macro EscribirMiNombre define las variables *document* y *dispatcher* del tipo *object*. Otros tipos comunes de variables son *string*, *integer* y *date*. Una tercera variable llamada *args1*, es una matriz (array) de valores de propiedades. Una variable de tipo *array* permite a una única variable contener múltiples valores, similar a almacenar múltiples página en un sólo libro. Los valores en

una matriz se numeran generalmente empezando por cero. El número entre paréntesis indica la posición más alta utilizable para almacenamiento. En este ejemplo solamente hay un valor y se numera cero.

Poniendo toda la macro junta

Los siguientes detalles son muy completos; no es importante entender todos los detalles en este momento. La primera línea define el principio de la macro.

```
sub EscribirMiNombre
```

Declara dos variables:

```
dim document as object
dim dispatcher as object
```

ThisComponent se refiere al documento actual.

La propiedad *CurrentController* de un documento se refiere a un servicio que “controla” el documento. Por ejemplo cuando usted escribe es el *CurrentController* quien lo nota. Entonces es el *CurrentController* quien muestra los cambios en la estructura del documento.

La propiedad *Frame* de un controlador devuelve la estructura principal de un documento. Por lo tanto, la variable llamada *document* se refiere a la estructura del documento, la cual recibe los comandos enviados.

```
document = ThisComponent.CurrentController.Frame
```

La mayoría de las tareas en LibreOffice se llevan a cabo enviando (dispatching) un comando. LibreOffice versión 2.0 introduce el servicio dispatch helper el cual hace la mayoría del trabajo para usar dispatches en macros. El método *CreateUnoService* acepta el nombre de un servicio e intenta crear una instancia de ese servicio. Al completarlo, la variable *dispatcher* contiene una referencia a un *DispatchHelper*.

```
dispatcher =
createUnoService("com.sun.star.frame.DispatchHelper")
```

Declara una matriz de propiedades. Cada propiedad tiene un nombre y un valor. En otras palabras, es un par nombre/valor. La matriz creada tiene una propiedad de índice cero.

```
dim args1(0) as new com.sun.star.beans.PropertyValue
```

Da a la propiedad el nombre “Text” y el valor “ Andrew Pitonyak”, que es el texto que se inserta cuando se ejecuta la macro.

```
args1(0).Name = "Text"
args1(0).Value = "Andrew Pitonyak"
```

Aquí es donde ocurre la magia. El dispatch helper envía un mensaje (dispatch) a la estructura del documento (almacenada en la variable llamada *document*) con el comando *.uno:InsertText*. Los siguientes dos argumentos, *frame name* y *search flags* van más allá del alcance de este documento. El último argumento es la matriz de valores de propiedades que será usado cuando se ejecute el comando *InsertText*.

```
dispatcher.executeDispatch(document, ".uno:InsertText", "", 0,
args1())
```

Finalmente, el final de la subrutina.

```
end sub
```

Crear una macro

Yo generalmente hago dos preguntas antes de grabar una macro:

- 1) ¿Se puede escribir la tarea como un conjunto simple de comandos?
- 2) ¿Se pueden ordenar los pasos de modo que el último comando deje el cursor preparado para el siguiente comando?

Un ejemplo complicado

Yo copio frecuentemente filas y columnas de datos desde un sitio web y los formateo como tabla en un documento de texto. Primero copio la tabla desde el sitio web al portapapeles. Para evitar formatos extraños y fuentes pego el texto en un documento de Writer como texto sin formato. Reformateo el texto con tabuladores entre las columnas de modo que puedo usar **Tabla** → **Convertir** → **Texto en tabla** para convertirlo en tabla.

Inspecciono el texto para ver si puedo grabar una macro para formatear el texto (recuerde las dos preguntas que hago). Como ejemplo copié el grupo de constantes FontWeight desde el sitio web de LibreOffice. La primera columna indica el nombre de la constante. Cada nombre va seguido de un espacio y un tabulador.

DONTKNOW	The font weight is not specified/known.
THIN	specifies a 50% font weight.
ULTRALIGHT	specifies a 60% font weight.
LIGHT	specifies a 75% font weight.
SEMILIGHT	specifies a 90% font weight.
NORMAL	specifies a normal font weight.
SEMIBOLD	specifies a 110% font weight.
BOLD	specifies a 150% font weight.
ULTRABOLD	specifies a 175% font weight.
BLACK	specifies a 200% font weight.

Quiero que la primera columna contenga el valor numérico, la segunda columna el nombre y la tercera la descripción. Lo que se desea es fácil llevarlo a cabo en todas las columnas excepto en DONTKNOW y NORMAL, las cuales no contienen un valor numérico —pero yo se que los valores son 0 y 100, por lo que los introduciré manualmente.

Los datos se pueden limpiar de múltiples maneras—todas ellas fáciles. El primer ejemplo utiliza escritura de teclado que asume que el cursor está al principio de la línea con el texto THIN .

- 1) Utilice **Herramientas** → **Macros** → **Grabar macro** para empezar a grabar.
- 2) Presione *Ctrl+Flecha derecha* para mover el cursor al comienzo de “specifies”.
- 3) Presione *Retroceso* dos veces para eliminar el tabulador y el espacio.
- 4) Presione *Tabulador* para añadir el tabulador sin el espacio detrás del nombre de la constante
- 5) Presione *Suprimir* para eliminar la s minúscula y presione *S* para añadir una S mayúscula.

- 6) Presione *Ctrl+Flecha derecha* dos veces para mover el cursor al inicio del número.
- 7) Presione *Ctrl+Mayúsculas+Flecha derecha* para seleccionar el número y mover el cursor delante del signo %.
- 8) Presione *Ctrl+C* para copiar la selección en el portapapeles.
- 9) Presione *Fin* para mover el cursor al final de la línea.
- 10) Presionar *Retroceso* dos veces para eliminar los dos espacios del final.
- 11) Presione *Inicio* para mover el cursor al principio de la línea.
- 12) Presione *Ctrl+V* para pegar el número seleccionado al principio de la línea.
- 13) Pegar el valor también pega un espacio extra, por eso presione *Retroceso* para remover el espacio extra.
- 14) Presione *Tabulador* para insertar un tabulador entre el número y el nombre.
- 15) Presione *Inicio* para moverse al principio de la línea.
- 16) Presione *Flecha abajo* para moverse a la siguiente línea.
- 17) Finalice la grabación y guarde la macro.

Cuesta mucho más tiempo leer y escribir los pasos que grabar la macro. Trabaje lentamente y piense en los pasos mientras los hace. Con práctica lo hará instintivamente.

La macro generada se ha modificado para contener el número de paso en los comentarios para enlazar el código con los pasos de arriba.

Listado 2: Copiar los valores numéricos al principio de la columna.

```

sub CopyNumToCol1
rem -----
rem define las variables
dim document as object
dim dispatcher as object
rem -----
rem obtiene acceso al documento
document = ThisComponent.CurrentController.Frame
dispatcher = createUnoService("com.sun.star.frame.DispatchHelper")

rem (2) Presionar Ctrl+Flecha derecha para mover el cursor al comienzo de
rem "specifies".
dispatcher.executeDispatch(document, ".uno:GoToNextWord", "", 0, Array())

rem (3) Presionar Retroceso dos veces para eliminar el tabulador y el
rem espacio.
dispatcher.executeDispatch(document, ".uno:SwBackspace", "", 0, Array())

rem -----
dispatcher.executeDispatch(document, ".uno:SwBackspace", "", 0, Array())

rem (4) Presionar Tabulador para añadir el tabulador sin el espacio detrás
rem del
rem nombre de la constante
dim args4(0) as new com.sun.star.beans.PropertyValue
args4(0).Name = "Text"
args4(0).Value = CHR$(9)

dispatcher.executeDispatch(document, ".uno:InsertText", "", 0, args4())

rem (5) Presionar Suprimir para eliminar la s minúscula...

```

```

dispatcher.executeDispatch(document, ".uno:Delete", "", 0, Array())

rem (5) ... y Presionar S para añadir una S mayúscula.
dim args6(0) as new com.sun.star.beans.PropertyValue
args6(0).Name = "Text"
args6(0).Value = "S"

dispatcher.executeDispatch(document, ".uno:InsertText", "", 0, args6())

rem (6) Presionar Ctrl+Flecha derecha dos veces para mover el cursor al
inicio
rem del número.
dispatcher.executeDispatch(document, ".uno:GoToNextWord", "", 0, Array())

rem -----
dispatcher.executeDispatch(document, ".uno:GoToNextWord", "", 0, Array())

rem (7) Presionar Ctrl+Mayúsculas+Flecha derecha para seleccionar el número.
dispatcher.executeDispatch(document, ".uno:WordRightSel", "", 0, Array())

rem (8) Presionar Ctrl+C para copiar la selección en el portapapeles.
dispatcher.executeDispatch(document, ".uno:Copy", "", 0, Array())

rem (9) Presionar Fin para mover el cursor al final de la línea.
dispatcher.executeDispatch(document, ".uno:GoToEndOfLine", "", 0, Array())

rem (10) Presionar Retroceso para eliminar los dos espacios del final.
dispatcher.executeDispatch(document, ".uno:SwBackspace", "", 0, Array())

rem -----
dispatcher.executeDispatch(document, ".uno:SwBackspace", "", 0, Array())

rem (11) Presionar Inicio para mover el cursor al principio de la línea.
dispatcher.executeDispatch(document, ".uno:GoToStartOfLine", "", 0, Array())

rem (12) Presionar Ctrl+V para pegar el número seleccionado al principio de
la línea.
dispatcher.executeDispatch(document, ".uno:Paste", "", 0, Array())

rem (13) Presionar Retroceso para remover el espacio extra.
dispatcher.executeDispatch(document, ".uno:SwBackspace", "", 0, Array())

rem (14) Presionar Tabulador para insertar un tabulador entre número y
nombre.
dim args17(0) as new com.sun.star.beans.PropertyValue
args17(0).Name = "Text"
args17(0).Value = CHR$(9)

dispatcher.executeDispatch(document, ".uno:InsertText", "", 0, args17())

rem (15) Presionar Inicio para moverse al principio de la línea.
dispatcher.executeDispatch(document, ".uno:GoToStartOfLine", "", 0, Array())

rem (16) Presionar Flecha abajo para moverse a la siguiente línea.
dim args19(1) as new com.sun.star.beans.PropertyValue
args19(0).Name = "Count"
args19(0).Value = 1

```

```
args19(1).Name = "Select"  
args19(1).Value = false
```

```
dispatcher.executeDispatch(document, ".uno:GoDown", "", 0, args19())  
end sub
```

Se usan movimientos del cursor para todas las operaciones (en oposición a la búsqueda). Si la ejecutamos en la línea DONTKNOW se mueve la palabra *weight* al principio de la línea y el primer "The" se cambia a "She". Esto no es perfecto, pero no debería haber ejecutado la macro en las líneas que no tienen el formato adecuado; necesitaría hacerlo manualmente.

Ejecutar la macro rápidamente

Es tedioso ejecutar la macro repetidamente utilizando **Herramientas** → **Macros** → **Ejecutar macro** (ver Figura 3). La macro puede ser ejecutada desde el IDE. Use **Herramientas** → **Macros** → **Organizar macros** → **LibreOffice Basic** para abrir el diálogo Macros Basic. Seleccione su macro y haga clic en **Editar** para abrir la macro en el IDE.

El IDE tiene un icono **Ejecutar programa BASIC** que ejecuta la primera macro en el IDE. A menos que lo cambie la primera macro es la macro vacía llamada Main. Modifique Main para que sea como se muestra en Listado 3.

Listado 3: Modifique Main para llamar a CopyNumToCol1.

```
Sub Main  
    CopyNumToCol1  
End Sub
```

Ahora puede ejecutar CopyNumToCol1 repetidamente haciendo clic en el icono Ejecutar programa BASIC en la barra de herramientas del IDE. Esto es muy rápido y fácil, especialmente para macros temporales que se van a usar unas pocas veces y después se van a descartar.

A veces la grabadora de macros falla

Entender los interiores de LibreOffice ayuda a entender cómo y por qué la grabadora de macros falla frecuentemente. El principal culpable está relacionado con el dispatch framework (marco de envío) y su relación con la grabadora de macros.

El dispatch framework

El propósito del dispatch framework es proveer acceso uniforme a los componentes (documentos) para los comandos que normalmente corresponden a elementos de menú. Puedo usar **Archivo** → **Guardar** en el menú, el atajo de teclado *Ctrl+G* o hacer clic en el icono **Guardar** de la barra de herramientas. Todos estos comandos son traducidos en el mismo "dispatch command" que se envía al documento actual.

El dispatch framework se puede usar también para enviar "comandos" de retorno a la IU (interfaz de usuario). Por ejemplo, después de guardar el documento el comando *Guardar archivo* se deshabilita. Tan pronto como se modifique el documento el comando *Guardar archivo* se habilita.

Si vemos un comando dispatch, es un texto como `.uno:InsertObject` o `.uno:GoToStartOfLine`. El comando se envía a la macro del documento y el macro pasa el comando hasta que encuentra un objeto que puede manejar el comando.

Cómo usa la grabadora de macros el dispatch framework

La grabadora de macros graba los dispatches generados. La grabadora es relativamente sencilla de implementar y el mismo comando que se envía se graba para uso posterior. El problema es que no todos los comandos enviados están completos. Por ejemplo, insertar un objeto genera el siguiente código:

```
dispatcher.executeDispatch(document, ".uno:InsertObject", "", 0, Array())
```

No es posible especificar que clase de objeto va a crear o insertar. Si un objeto se inserta desde un archivo, usted no puede especificar qué archivo insertar.

Yo grabé una macro y usé **Herramientas** → **Opciones** para abrir y modificar elementos de configuración. La macro generada no grabó ningún cambio, de hecho, el código generado está comentado por lo que ni siquiera se ejecuta.

```
rem dispatcher.executeDispatch(document,
    ".uno:OptionsTreeDialog", "", 0, Array())
```

Si se abre un diálogo, el comando para abrir el diálogo probablemente sea generado. Cualquier trabajo hecho dentro del diálogo generalmente no se graba. Algunos ejemplos incluyen los diálogos de organización de macros, insertar caracteres especiales y diálogos similares. Otros problemas posibles usando la grabadora de macros incluyen cosas como insertar una fórmula, establecer área de usuario, establecer filtros en Calc, acciones en formularios de bases de datos y exportar un documento a un archivo PDF encriptado. Sin embargo nunca sabrá con certeza cuál funcionará a menos que lo intente. Por ejemplo, las acciones del diálogo buscar se capturan correctamente.

Otras opciones

Cuando la grabadora de macros no es capaz de resolver un problema específico, la solución normal es escribir código utilizando los objetos de LibreOffice.

Desafortunadamente la curva de aprendizaje de los objetos de LibreOffice es muy empinada. Generalmente es mejor empezar con ejemplos simples e ir diversificando lentamente a medida que vaya aprendiendo.

Si graba macros de Calc y la grabadora puede generar correctamente una macro, hay un complemento creado por Paolo Mantovani, el cual convierte las macros de Calc cuando se graban. El código final manipula los objetos de LibreOffice en vez de generar dispatches. Esto puede ser muy útil para aprender el modelo de objetos.

Puede descargar la grabadora de macros del sitio web de Paolo directamente o del sitio web de macros de LibreOffice. Debería comprobar ambos sitios para ver cual tiene la última versión.

<http://www.paolo-mantovani.org/downloads/DispatchToApiRecorder/>

<http://www.libreofficemacros.org/user.php>

Organización de macros

En LibreOffice las macros se agrupan en módulos, los módulos se agrupan en bibliotecas y las bibliotecas se agrupan en contenedores de bibliotecas. Una biblioteca se usa generalmente como grupo principal para una categoría completa de macros o una aplicación completa. Los módulos generalmente dividen la funcionalidad, tales como interacción con el usuario y cálculos. Las macros individuales son subrutinas y funciones.

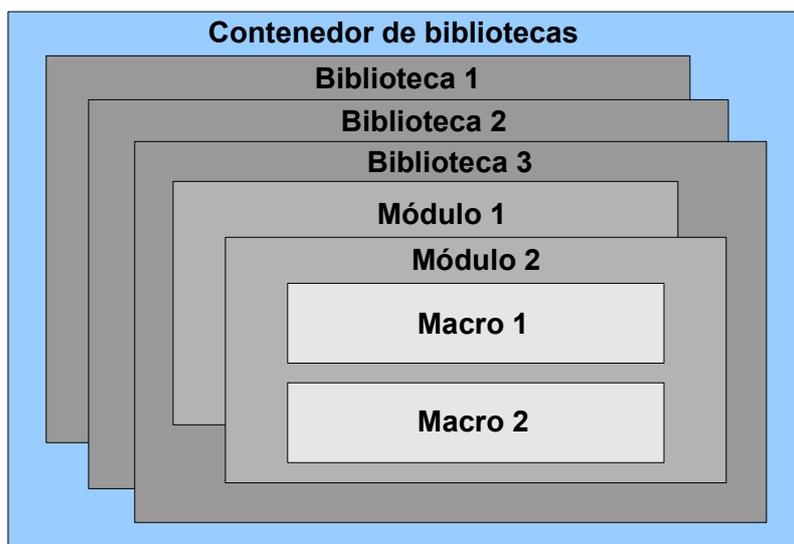


Figura 4: Jerarquía de bibliotecas de macros

Un informático usaría la Figura 5 para describir precisamente la situación. El texto “1..*” significa uno o más y “0..*” significa cero o más. El rombo negro significa está compuesto o contiene.

- Un contenedor de bibliotecas contiene una o más bibliotecas y cada biblioteca está contenida en un contenedor.
- Una biblioteca contiene cero o más módulos y cada módulo está contenido en una biblioteca.
- Un módulo contiene cero o más macros y cada macro está contenida en un módulo.

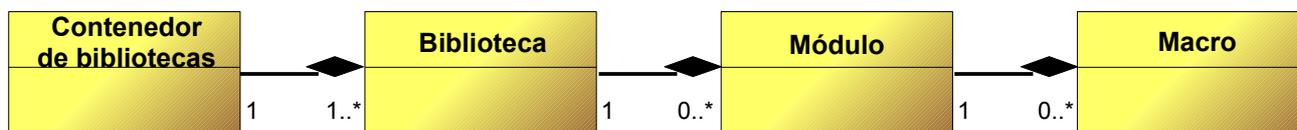


Figura 5: Jerarquía de bibliotecas de macros

Utilice **Herramientas** → **Macros** → **Organizar macros** → **LibreOffice Basic** para abrir el diálogo Macros Basic de LibreOffice (ver Figura 6). Todos los contenedores de bibliotecas disponibles se muestran en la lista *Desde la macro*. Cada documento es un contenedor de bibliotecas, capaz de contener múltiples bibliotecas. La aplicación en sí misma actúa como dos contenedores de bibliotecas, un contenedor para macros distribuidas con LibreOffice llamado Macros de LibreOffice y un contenedor para macros personales llamado Mis macros. Como se puede ver en la Figura 6, solamente hay abiertos dos documentos en este momento.

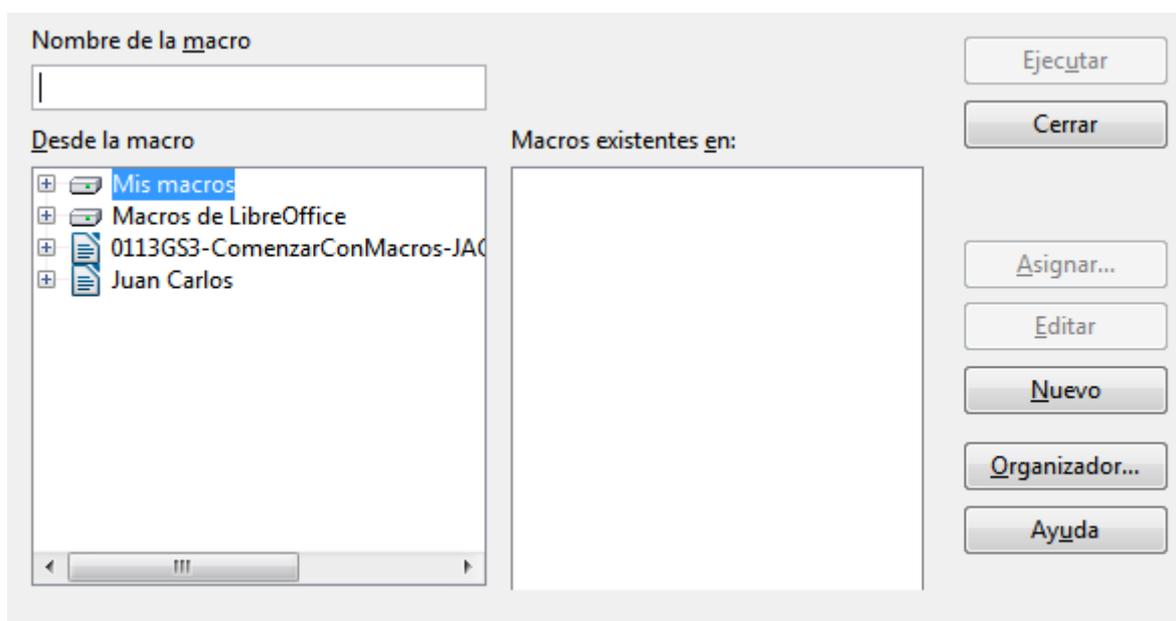


Figura 6: Los contenedores de bibliotecas se muestran a la izquierda

Las macros de LibreOffice se almacenan con el código de tiempo de ejecución de la aplicación, el cuál no se puede editar a menos que sea administrador. Esto es así porque estas macros no deberían modificarse y usted no debería almacenar sus propias macros en el contenedor de LibreOffice.

A menos que sus macros se apliquen a un único documento y solo a ese documento, sus macros deberían seguramente almacenarse en el contenedor Mis macros. El contenedor mis macros se almacena en su área de usuario o directorio personal.

Si una macro está contenida en un documento, la macro grabada intentará ejecutarse en ese documento, principalmente porque la macro usa "ThisComponent" para sus acciones.

Cada contenedor de bibliotecas contiene una biblioteca llamada *Standard*. Es mejor que cree sus propias bibliotecas con nombres significativos que usar la biblioteca Standard. No solamente los nombres significativos son más fáciles de manejar, sino que se pueden importar en otros contenedores de bibliotecas mientras que en la biblioteca Standard no se puede.

Precaución



LibreOffice le permite importar bibliotecas en un contenedor de bibliotecas, pero no le permite sobrescribir la biblioteca llamada Standard. Por lo tanto, si almacena sus macros en la biblioteca Standard no la podrá importar desde otro contenedor de bibliotecas.

Del mismo modo que es sensato dar nombres significativos a sus bibliotecas, es prudente utilizar nombres significativos para los módulos. Por defecto LibreOffice usa nombres como Module1. Siéntase libre de utilizar sus propios nombres significativos.

Cuando crea sus macros debe decidir dónde almacenarlas. Almacenar una macro en un documento es útil si el documento va a ser compartido y quiere que la macro sea incluida con el documento. Las macros almacenadas se encuentran en el contenedor de bibliotecas llamado Mis macros, sin embargo, están disponibles globalmente para todos los documentos.

Las macros no están disponibles hasta que se carga la biblioteca que las contiene. La biblioteca Standard y la biblioteca de plantilla, sin embargo, se cargan automáticamente. Una biblioteca cargada se muestra de modo diferente a como la hace una que no está cargada. Para cargar una biblioteca y los módulos que contiene haga doble clic en la biblioteca.

¿Dónde se almacenan las macros?

LibreOffice almacena los datos específicos del usuario en un directorio bajo el directorio particular del usuario. Por ejemplo, en Windows es C:\Documents and Settings\\Application Data.

Las macros de usuario se almacenan en LibreOffice\3\user\basic. Cada biblioteca se almacena en su propio directorio del directorio basic.

No es importante aprender donde se almacenan las macros para un uso ocasional. Sin embargo si sabe donde se almacenan puede crear una copia de seguridad, compartir sus macros o inspeccionarlas si hay algún error. Por ejemplo en algunas de mis actualizaciones de LibreOffice, todas mis macros desaparecieron. Aunque las macros estaban aún en el disco no se copiaron a los nuevos directorios. La solución fue importar las macros en la nueva instalación.

Utilice **Herramientas** → **Macros** → **Organizar diálogos** para abrir el diálogo Organizador de macros. Otro método corriente de abrir este diálogo es usando **Herramientas** → **Macros** → **Organizar macros** → **LibreOffice Basic** para abrir el diálogo Macros Basic de LibreOffice y hacer clic sobre el botón **Organizador** (ver Figura 7)

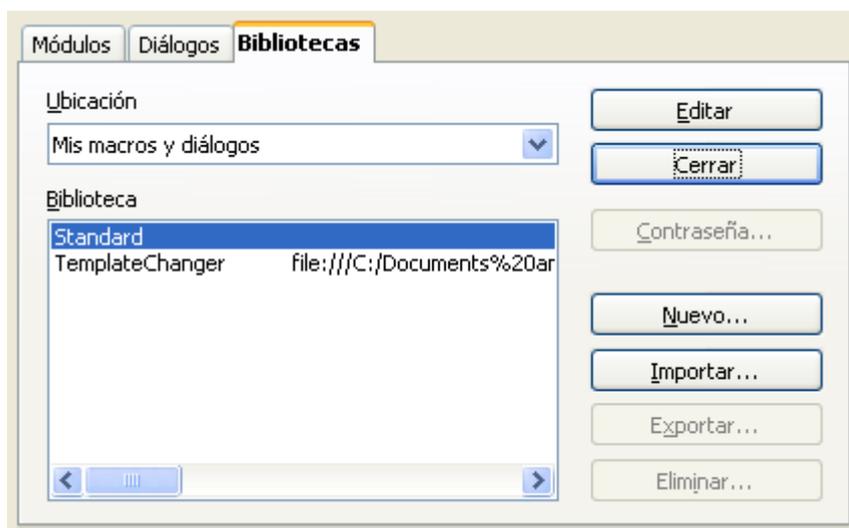


Figura 7: El diálogo organizador de macros

Importar macros

El diálogo Organizador de macros de LibreOffice proporciona funcionalidad para crear, eliminar y renombrar bibliotecas, módulos y diálogos. Seleccione el contenedor de bibliotecas que va a usar y haga clic en el botón **Importar** para importar bibliotecas de macros (vea Figura 8)

Sugerencia

No puede importar la biblioteca llamada Standard.

Sugerencia

En Linux, los archivos específicos de LibreOffice se almacenan en un directorio cuyo nombre comienza por punto. Directorios y archivos cuyo nombre comienza por punto no se muestran en un diálogo normal de selección. Para abrir el directorio yo navego hasta el directorio padre, escribo el nombre .libreoffice y hago clic en **Abrir**. Esto abre el directorio que inicialmente no se mostraba.

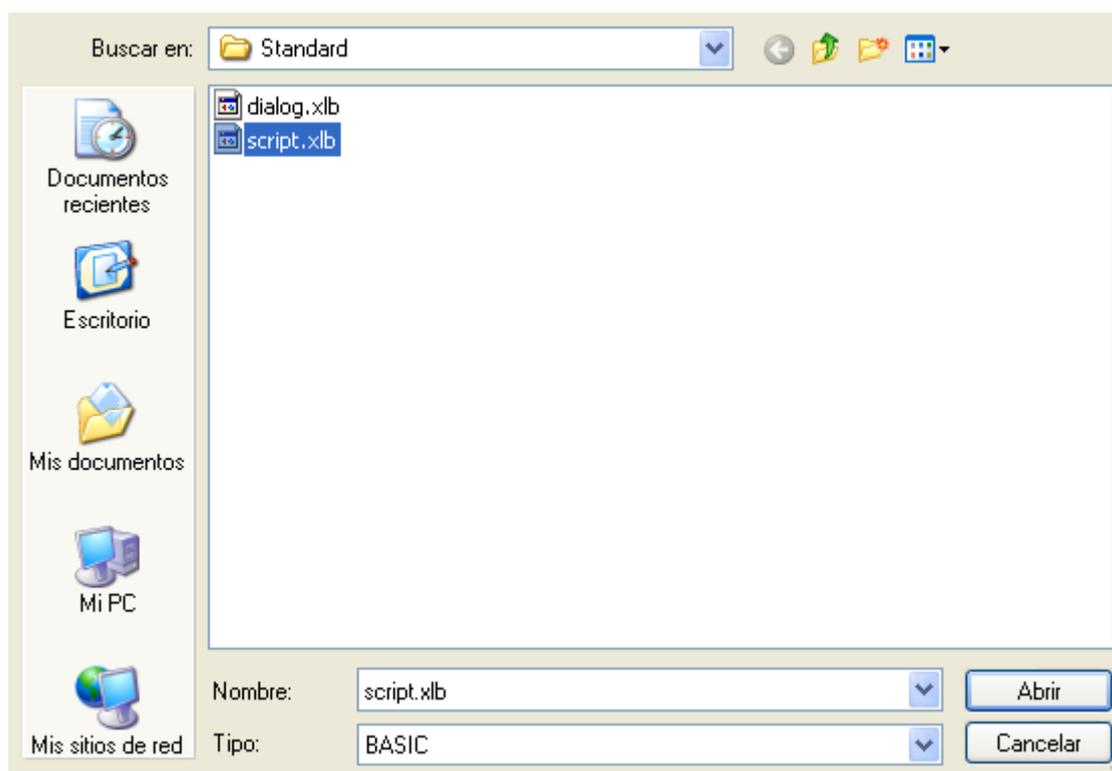


Figura 8: Seleccionar una biblioteca de macros para importar

Navegar hasta el directorio que contiene la biblioteca a importar. Generalmente hay dos archivos para escoger, dialog.xlb y script.xlb. No importa cual de estos archivos seleccione, se importarán los dos. Seleccione un archivo y pulse **Abrir** para continuar (vea Figura 9)

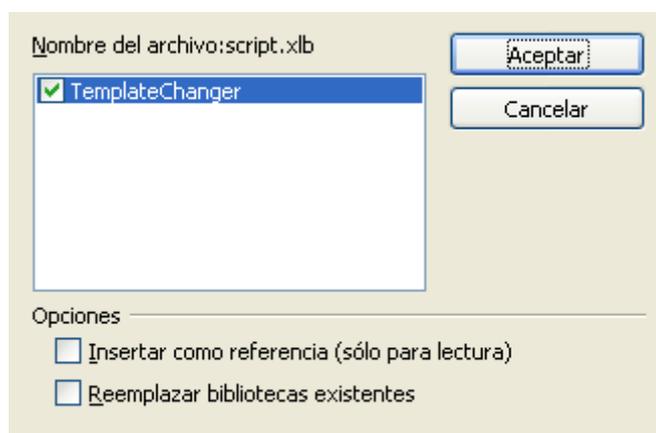


Figura 9: Seleccione las opciones de importar

Si la biblioteca ya existe, no se reemplazará a menos que esté marcado **Reemplazar bibliotecas existentes**. Si está marcado **Insertar como referencia** la biblioteca se referencia en su localización actual pero no puede modificarla. En cambio, si **Insertar como referencia** no está marcado la biblioteca se copia al directorio de macros del usuario.

Las macros se pueden almacenar en bibliotecas dentro de documentos de LibreOffice. Seleccione un documento en lugar de un directorio del disco (como se muestra en la Figura 8) para importar bibliotecas contenidas en un documento.

Descargar macros para importar

Hay macros disponibles para descargar. Algunas macros están contenidas en documentos, otras son archivos normales que debe seleccionar e importar y otras son texto de macros para copiar y pegar en el IDE de Basic; utilice **Herramientas** → **Macros** → **Organizar macros** → **LibreOffice Basic** para abrir el diálogo Macros Basic de LibreOffice, elija la macro a editar y haga clic en **Editar** para abrir el IDE de Basic.

Algunas macros están disponibles como descarga gratuita en Internet

Tabla 1. Sitios donde encontrar ejemplos de macros

Localización	Descripción
http://www.oocomacs.org/	Excelente colección de paquetes de macros.
http://www.pitonyak.org/oo.php	Material de referencia relacionado con macros.
http://www.pitonyak.org/database/	Material de referencia relacionado con macros de bases de datos.
http://development.openoffice.org/	Montones de enlaces a todo.
http://www.ooforum.org/	Muchos ejemplos y ayuda.
http://user.services.openoffice.org/	Otro foro con muchos ejemplos y ayuda.

Cómo ejecutar una macro

Un método común de ejecutar una macro es el siguiente:

- 1) Utilice **Herramientas** → **Macros** → **Ejecutar macro** para abrir el diálogo Selector de macro (vea Figura 10)
- 2) Seleccione la biblioteca y el módulo en la lista Biblioteca (lado izquierdo).
- 3) Seleccione la macro en la lista Nombre de macro (lado derecho).
- 4) Haga clic en **Ejecutar** para ejecutar la macro.

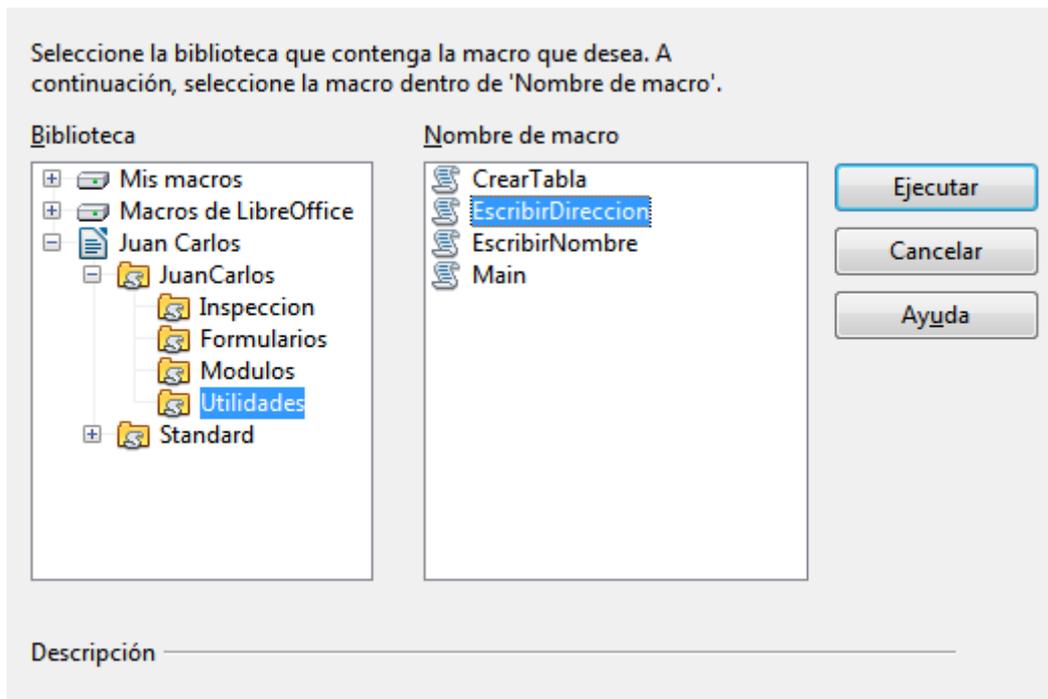


Figura 10: Utilice el diálogo Selector de macros para ejecutar las macros

Aunque puede utilizar **Herramientas** → **Macros** → **Ejecutar macros** para ejecutar una macro, esto no es eficiente para ejecutar macros frecuentemente. Una técnica más común es asignar una macro a un botón de una barra de herramientas, a un elemento de menú, a un atajo de teclado o a un botón insertado en un documento. Mientras selecciona un método es bueno preguntarse cuestiones como:

- ¿Debe estar la macro disponible para un solo documento o globalmente para todos los documentos?
- ¿Está la macro relacionada con un tipo específico de documento, como un documento de Calc?
- ¿Con qué frecuencia se usará la macro?

Las respuestas determinarán dónde almacenar la macro y cómo estará disponible. Por ejemplo, seguramente no añadirá una macro raramente usada a una barra de herramientas. Para ayudarle a determinar sus elecciones vea la Tabla 2

Tabla 2. Métodos para iniciar una macro

Tipo	LibreOffice	Tipo de documento	Documento
Barra de herramientas	No	Sí	Sí
Menú	No	Sí	Sí
Atajo de teclado	Sí	Sí	No
Evento	Sí	No	Sí

Para añadir un elemento de menú, atajo de teclado o icono de una barra de herramientas que llame a una macro utilice el diálogo Personalizar (vea Figura 12). Abra este diálogo de cualquiera de estas formas:

- Elija **Herramientas** → **Personalizar** en la barra de menús.

- Cada barra de herramientas tiene un icono  que abre un menú; elija la opción **Personalizar barra de herramientas**.

Sugerencia

Una cobertura completa del diálogo Personalizar va más allá del alcance de este documento. Haga clic en el botón **Ayuda** para acceder a las páginas de ayuda incluidas en LibreOffice.

El menú personalizar contiene pestañas para configurar menús, atajos de teclado, barras de herramientas y eventos.

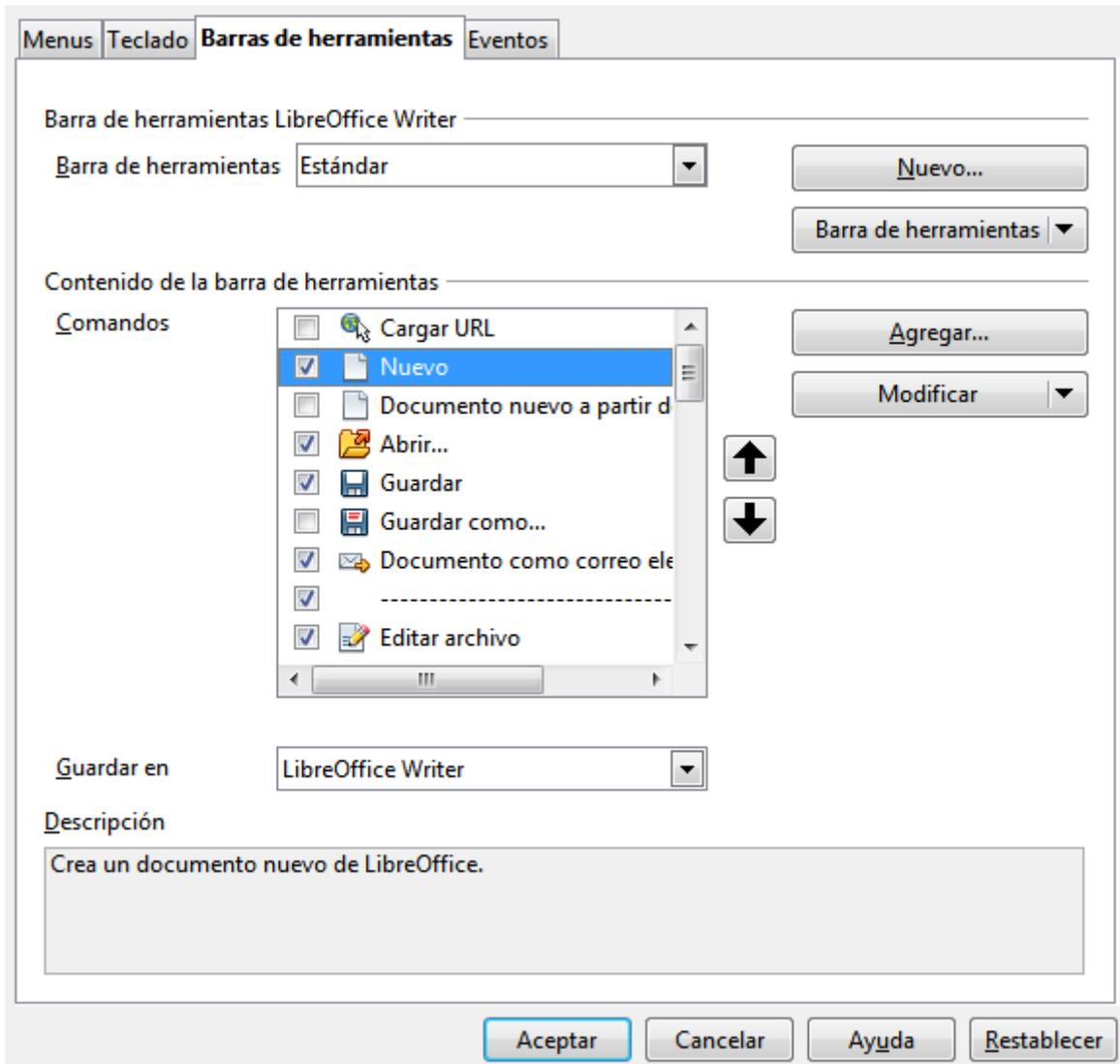


Figura 11: Diálogo Personalizar de LibreOffice

Barra de herramientas

Se pueden añadir macros a las barras de herramientas. Para más información de cómo modificar barras de herramientas vea el capítulo 14 (Personalizar LibreOffice).

Elemento de menú

Utilice **Herramientas** → **Personalizar** para abrir el diálogo Personalizar y seleccione la pestaña Menús. Puede modificar un menú existente o crear nuevos menús que llamen a macros. Para saber más acerca de modificar menús vea el capítulo 14.

Atajos de teclado

Utilice **Herramientas** → **Personalizar** para abrir el diálogo personalizar y seleccione la pestaña Teclado. Cómo asignar atajos de teclado se discute en el capítulo 14.

Evento

En LibreOffice, cuando sucede algo decimos que ha ocurrido un evento. Por ejemplo, se abre un documento, se presiona una tecla o se mueve el ratón. LibreOffice permite que los eventos llamen a macros; se dice entonces que la macro es un manipulador de evento. Una cobertura completa de los manipuladores de eventos está muy por encima del alcance de este documento, pero un poco de conocimiento puede ser útil.

Precaución



Sea cuidadoso cuando configure un manipulador de eventos. Por ejemplo, supongamos que escribe un evento que se llama cada vez que se presiona una tecla, pero comente un error y el evento no se manipula adecuadamente. Un resultado posible es que su evento consuma todas las pulsaciones del teclado obligándole a cerrar LibreOffice a la fuerza.

Utilice **Herramientas** → **Personalizar** para abrir el diálogo Personalizar y seleccione la pestaña Eventos (vea Figura 11). Los eventos en el diálogo Personalizar están relacionados con la aplicación completa o con un documento específico. Utilice la lista desplegable “Guardar en” para elegir LibreOffice o un documento específico.

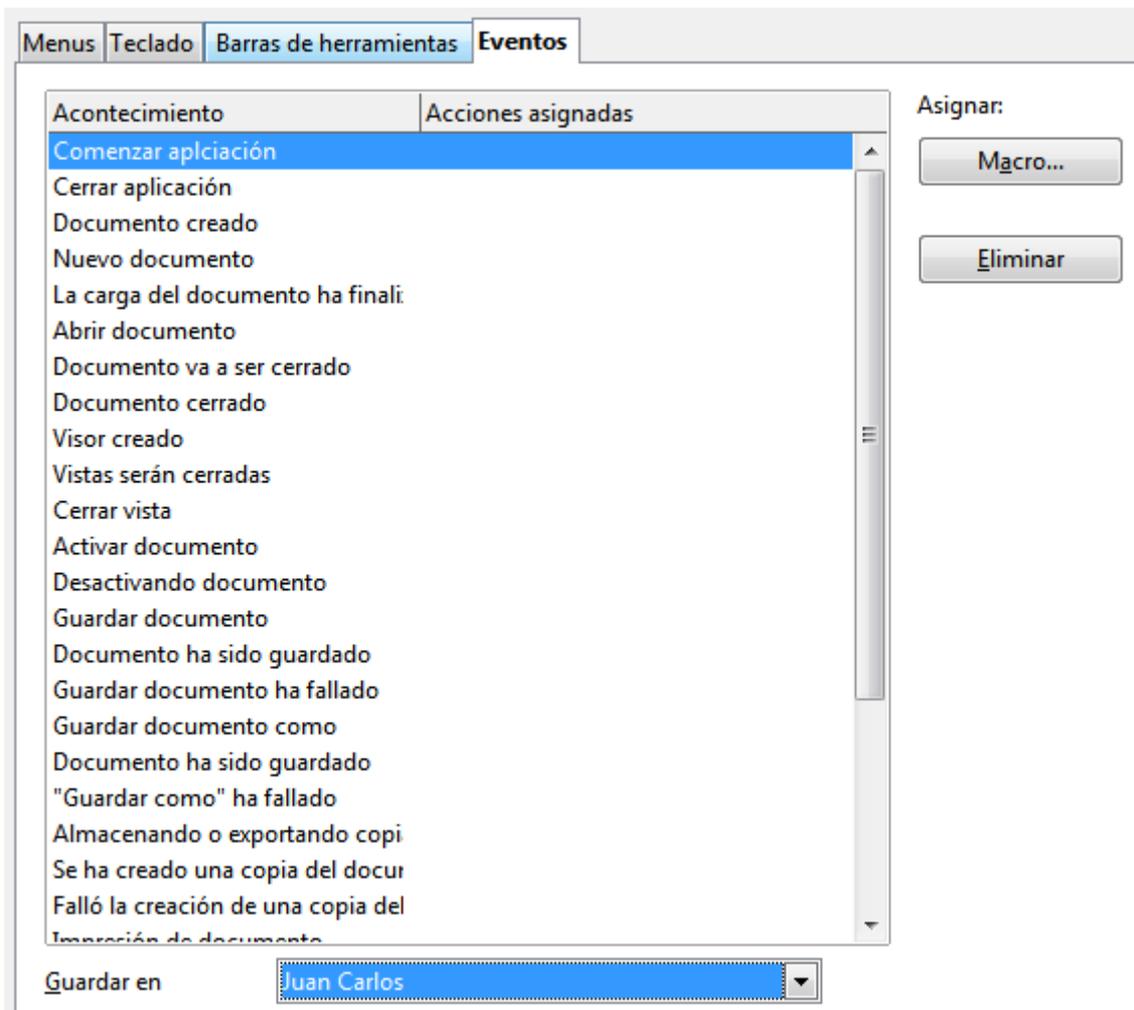


Figura 12: Asignar macro a una aplicación a nivel de evento

Un uso común es asignar el evento Abrir documento para llamar a una macro específica. La macro entonces realiza ciertas tareas de configuración del documento. Seleccione el evento deseado y haga clic en el botón **Macro** para abrir el diálogo Selector de macro (vea Figura 13)

Seleccione la macro deseada y haga clic en **Aceptar** para asignar la macro al evento. La pestaña Eventos muestra que el evento se ha asignado a una macro (ver Figura 14). Cuando el documento se abre se ejecuta la macro EscribirHola.

Muchos objetos en un documento se pueden configurar para que llamen a macros cuando ocurra un evento. El uso más común es añadir un control, como un botón, en un documento. Incluso hacer doble clic en un gráfico abre un diálogo con la pestaña Macros para que pueda asignar una macro a un evento.

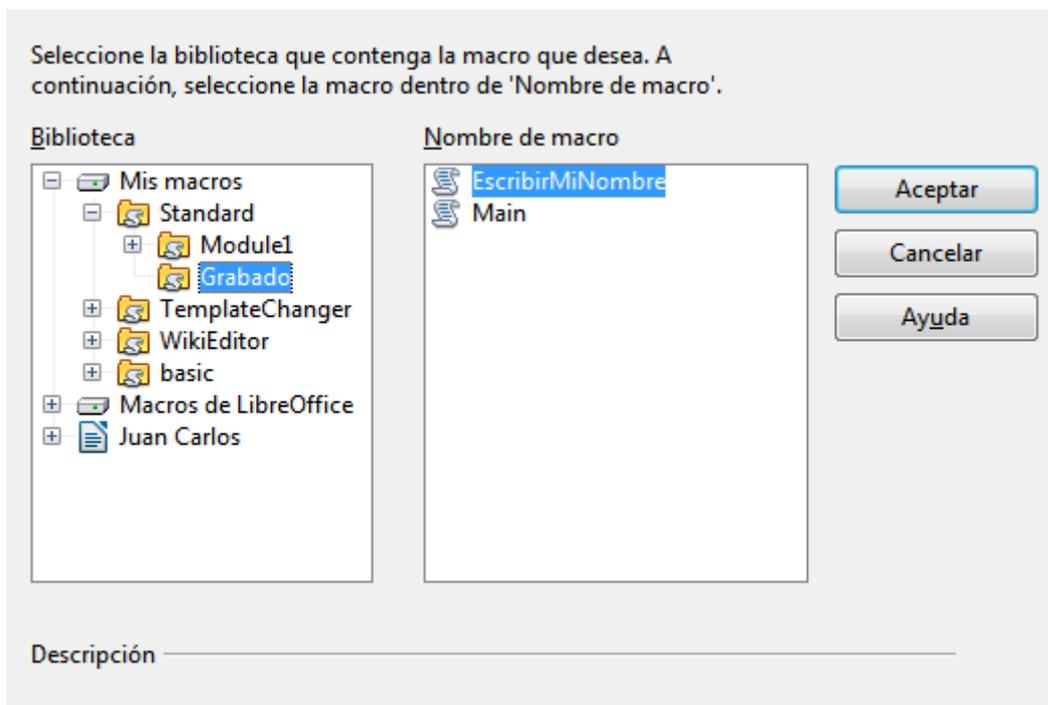


Figura 13: Asignar una macro al evento Abrir documento

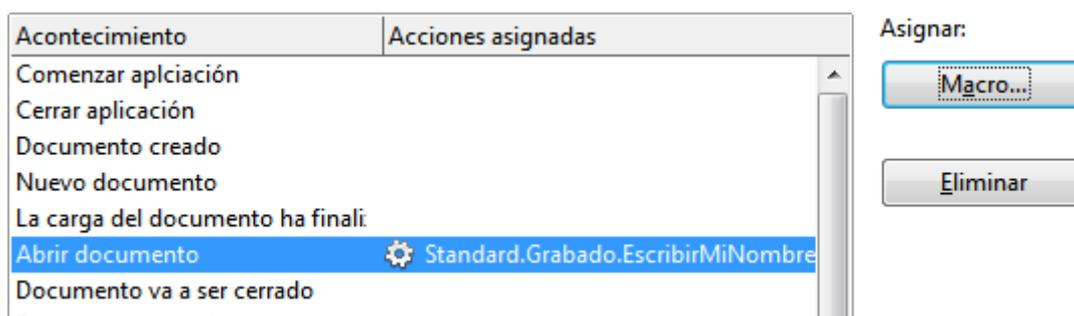


Figura 14: Se ha asignado EscribirHola al evento Abrir documento

Extensiones

Una extensión es un paquete que se puede instalar en LibreOffice para añadir una nueva funcionalidad. Las extensiones se pueden escribir en casi cualquier lenguaje de programación y puede ser simple o sofisticada. Las extensiones se pueden agrupar en los tipos:

- Complementos de Calc, que proveen nuevas funcionalidades para Calc, incluso nuevas funciones que actúan como las funciones integradas normales.
- Nuevos componentes y funcionalidad, las cuales normalmente incluyen algún nivel de integración en la IU, como nuevos menús o barras de herramientas.
- Pilotos de datos que se usan directamente en Calc
- Complementos de Chart con nuevos tipos de gráficos.
- Componentes lingüísticos como revisores de ortografía.
- Plantillas de documento e imágenes.

Aunque las extensiones individuales se pueden encontrar en diferentes lugares hay un repositorio de extensiones en <http://extensions.libreoffice.org/extension-center>. Para saber más acerca de cómo obtener e instalar extensiones vea el capítulo 14 (Personalizar LibreOffice).

Escribir macros sin la grabadora

Los ejemplos cubiertos en este capítulo han sido creados usando la grabadora de macros y el dispatcher. Puede escribir macros que accedan directamente a los objetos comprendidos en LibreOffice. En otras palabras, puede manipular directamente un documento.

Manipular directamente los objetos internos de LibreOffice es un tema avanzado que va más allá del alcance de este capítulo. Sin embargo un ejemplo sencillo demuestra como funciona esto.

Listado 4: Añadir el texto "Hola" al documento actual.

```
Sub AppendHello
  Dim oDoc
  Dim sTextService$
  Dim oCurs

  REM ThisComponent se refiere al documento activo actualmente.
  oDoc = ThisComponent

  REM Verifica que esto es un documento de texto
  sTextService = "com.sun.star.text.TextDocument"
  If NOT oDoc.supportsService(sTextService) Then
    MsgBox "Esta macro solo funciona en documentos de texto"
    Exit Sub
  End If

  REM Obtiene la vista del cursor del controlador actual.
  oCurs = oDoc.currentController.getViewCursor()

  REM Mueve el cursor hasta el final del documento.
  oCurs.gotoEnd(False)

  REM Inserta el texto "Hola" al final del documento.
  oCurs.Text.insertString(oCurs, "Hola", False)
End Sub
```

Encontrar más información

Hay disponibles numerosos recursos que proporcionan ayuda para escribir macros. Utilice **Ayuda** → **Ayuda de LibreOffice** para abrir las páginas de ayuda de LibreOffice. La esquina superior izquierda del sistema de ayuda de LibreOffice tiene una lista desplegable que determina que grupo de la ayuda se muestra. Para ver la ayuda para Basic seleccione LibreOffice *Basic* en la lista.

Material incluido

Con LibreOffice se incluyen muchas macros excelentes. Utilice **Herramientas** → **Macros** → **Organizar macros** → **LibreOffice Basic** para abrir el diálogo de Macros. Extienda la biblioteca Tools en el contenedor de bibliotecas LibreOffice. Inspeccione el módulo Debug —algunos buenos ejemplos son WritedbgiInfo (documento) y printdbgInfo (hoja de cálculo).

Recursos online

Los siguientes enlaces y referencias continen información acerca de programación de macros:

<http://user.services.openoffice.org/> (Foros de OpenOffice.org, muy bien soportado; los voluntarios también contestan preguntas de LibreOffice)

<http://api.openoffice.org/docs/common/ref/com/sun/star/module-ix.html> (Referencia oficial de la IDL; aquí encontrará la descripción de casi cualquier comando)

http://wiki.services.openoffice.org/wiki/Documentation/BASIC_Guide (guía oficial de programación con LibreOffice BASIC)

http://wiki.services.openoffice.org/wiki/Documentation/DevGuide/LibreOffice_Developers_Guide (Guía oficial de desarrolladores de LibreOffice; contiene una explicación detallada)

<http://www.pitonyak.org/oo.php> (Página de macros de Andrew Pitonyak)

<http://www.pitonyak.org/AndrewMacro.odt> (Numerosos ejemplos de trabajar con macros)

<http://www.pitonyak.org/book/> (Libro sobre macros de Andrew Pitonyak)

<http://www.pitonyak.org/database/> (numerosas macros de ejemplo utilizando Base)

<http://docs.sun.com/app/docs/doc/819-0439> (Libro de Sun acerca de la programación de macros —muy bien escrito y ordenado; la guía de programación de LibreOffice BASIC y la Guía de desarrolladores de LibreOffice derivan de este libro).

Material impreso y eBooks

Los siguiente libros están disponibles para la adquisición a sus editores tanto en impreso como en eBook.

OpenOffice.org Macros Explained de Andrew Pitonyak. Vea <http://www.hentzenwerke.com/catalog/oome.htm>.

Learn OpenOffice.org Spreadsheet Macro Programming de Dr. Mark Alexander Bain. Vea <http://www.packtpub.com/libreoffice-libreofficebasic-calc-automation/book>.