



**LibreOffice**  
The Document Foundation

Erste Schritte

*Kapitel 13*  
*Einführung in Makros*

## Copyright

---

Dieses Dokument unterliegt dem Copyright © 2011. Die Beitragenden sind unten aufgeführt. Sie dürfen dieses Dokument unter den Bedingungen der GNU General Public License (<http://www.gnu.org/licenses/gpl.html>), Version 3 oder höher, oder der Creative Commons Attribution License (<http://creativecommons.org/licenses/by/3.0/>), Version 3.0 oder höher, verändern und/oder weitergeben.

Warennamen werden ohne Gewährleistung der freien Verwendbarkeit benutzt.

Fast alle Hardware- und Softwarebezeichnungen und weitere Stichworte und sonstige Angaben, die in diesem Buch verwendet werden, sind als eingetragene Marken geschützt.

Da es nicht möglich ist, in allen Fällen zeitnah zu ermitteln, ob ein Markenschutz besteht, wird das Symbol (R) in diesem Buch nicht verwendet.

## Mitwirkende/Autoren

Ralf Albrecht  
Nino Novak  
Florian Reisinger

Erhardt Balthasar  
Jochen Schiffers

Christian Kühn  
Klaus-Jürgen Weghorn

## Englisches Originaldokument

Ron Faile Jr.

## Rückmeldung (Feedback)

Kommentare oder Vorschläge zu diesem Dokument können Sie in deutscher Sprache an die Adresse [discuss@de.libreoffice.org](mailto:discuss@de.libreoffice.org) senden.

### Vorsicht



Alles, was an eine Mailingliste geschickt wird, inklusive der E-Mail-Adresse und anderer persönlicher Daten, die die E-Mail enthält, wird öffentlich archiviert und kann nicht gelöscht werden. Also, schreiben Sie mit Bedacht!

## Danksagung

Das englische Original-Dokument basiert auf Kapitel 13 des „Getting Started with OpenOffice.org 3.3“. Autoren dieses Kapitels sind:

Andrew Pitonyak

Jean Hollis Weber

## Datum der Veröffentlichung und Softwareversion

Veröffentlicht am 22.04.2012. Basierend auf der LibreOffice Version 3.3.

## Anmerkung für Macintosh Nutzer

---

Einige Tastenbelegungen (Tastenkürzel) und Menüeinträge unterscheiden sich zwischen der Macintosh Version und denen für Windows- und Linux-Rechnern. Die unten stehende Tabelle gibt Ihnen einige grundlegende Hinweise dazu. Eine ausführlichere Aufstellung dazu finden Sie in der Hilfedatei des jeweiligen Module.

<b>Windows/Linux</b>	<b>entspricht am Mac</b>	<b>Effekt</b>
Menü-Auswahl <b>Extras</b> → <b>Optionen</b>	LibreOffice → Einstellungen	Zugriff auf die Programmooptionen
Rechts-Klick	Control+Klick	Öffnen eines Kontextmenüs
Ctrl (Control) oder Strg (Steuerung)	⌘ ( <i>Command</i> )	Tastenkürzel in Verbindung mit anderen Tasten
F5	Shift+⌘+F5	öffnet den Dokumentnavigator Dialog
F11	⌘+T	öffnet den Formatvorlagen Dialog

## Inhalt

---

<i>Anmerkung für Macintosh Nutzer</i> .....	2
<i>Ihr erstes Makro</i> .....	5
<i>Ein vorhandenes Makro hinzufügen</i> .....	5
<i>Ein einfaches Makro selbst erstellen</i> .....	7
<i>Makro ausführen</i> .....	8
<i>Makro-Code ansehen und bearbeiten</i> .....	9
<i>Kommentare starten mit REM</i> .....	10
<i>Subroutinen mit SUB festlegen</i> .....	10
<i>Variablendefinition mittels DIM</i> .....	10
<i>Das Makro im Zusammenhang</i> .....	11
<i>Makro erstellen</i> .....	12
<i>Die integrierte Entwicklungsumgebung</i> .....	12
<i>Schnelles Ausführen eines Makros</i> .....	16
<i>Bekannte Probleme mit dem Makrorekorder</i> .....	16
<i>Dispatch Framework: was ist das?</i> .....	17
<i>Wie der Makrorekorder den Dispatch Framework verwendet</i> .....	17
<i>Weitere Möglichkeiten</i> .....	17
<i>Makroorganisation</i> .....	18
<i>Wo werden Makros gespeichert?</i> .....	19
<i>Makros importieren</i> .....	20
<i>Makros für den Import herunterladen</i> .....	22
<i>Wie man ein Makro ausführt</i> .....	23
<i>Symbolleiste</i> .....	25
<i>Menüeintrag</i> .....	25
<i>Tastaturkürzel</i> .....	25
<i>Ereignis</i> .....	26
<i>Erweiterungen (Extensions)</i> .....	28
<i>Makros schreiben ohne Recorder</i> .....	29
<i>Weitere Informationen</i> .....	29
<i>In LibreOffice enthaltene Makros</i> .....	29
<i>Onlinequellen</i> .....	30

## Ihr erstes Makro

Ein Makro ist eine gespeicherte Abfolge von Kommandos oder Tastatureingaben, auf die für eine spätere Verwendung wieder zugegriffen werden kann.

Die LibreOffice-Makrosprache ist sehr flexibel und erlaubt die Automatisierung sowohl einfacher als auch komplexer Aufgaben. Makros sind immer dann hilfreich, wenn Sie eine Aufgabe immer wieder auf die gleiche Art und Weise erledigen müssen oder wenn Sie eine Aufgabe auf Knopfdruck abarbeiten möchten, für die Sie eigentlich mehrere Schritte benötigen würden.

Die LibreOffice-Makrosprache basiert auf der Programmiersprache BASIC - meist „LibreOffice Basic“ oder einfach nur „Basic“ genannt. Weil LibreOffice Basic naturgemäß über sehr viele Programmfunktionen verfügt, ist Basic recht komplex und damit zu Beginn recht lernintensiv. Um Sie in die Nutzung von Makros einzuführen, schlagen wir Ihnen daher vor, erst mal mit der Ausführung von fertigen Makros zu beginnen, die von Anderen erstellt wurden, oder den eingebauten Makro-Rekorder zu nutzen, mit welchem Sie Tastaturbefehle aufzeichnen und als Makro für eine spätere Nutzung speichern können.

Derzeit wird der Makro-Rekorder in LibreOffice als eines „experimentelles“ Programmfeature eingestuft. Um den Makro-Rekorder nutzen zu können, müssen Sie diesen daher unter **Extras** → **Optionen...** → **LibreOffice** → **Allgemein** die Option „Aktiviere Experimentelle Funktionen“ aktivieren.

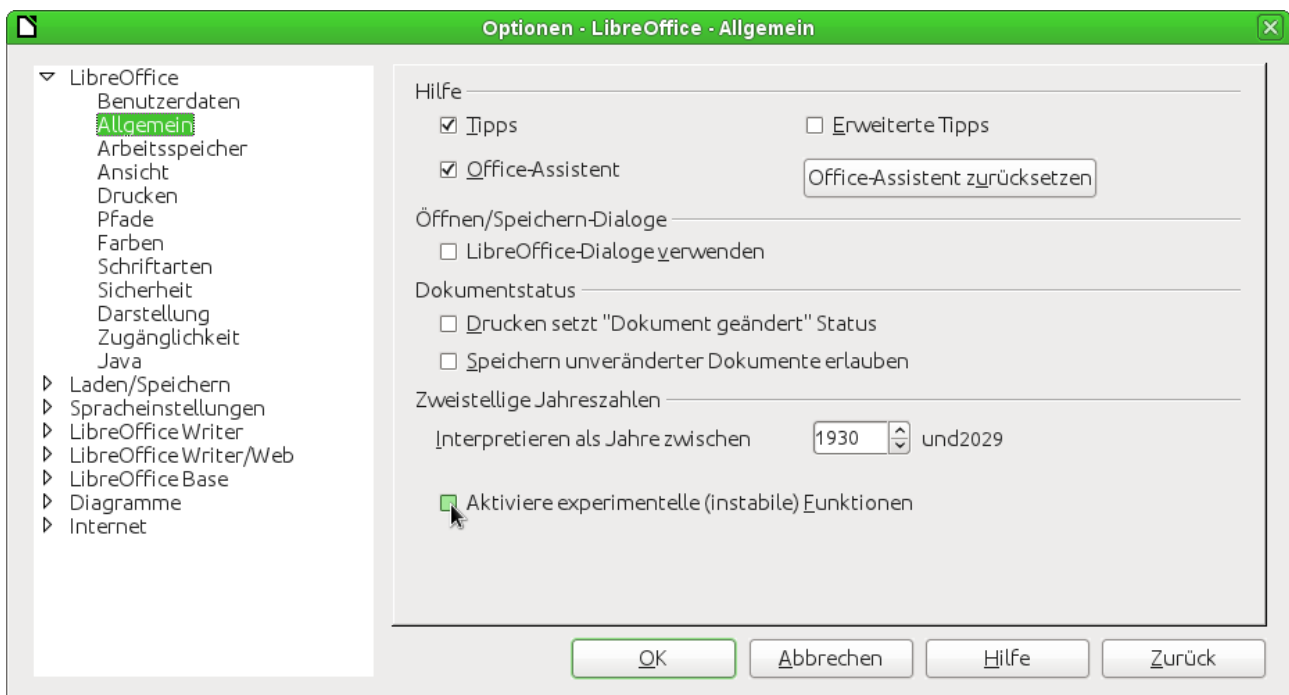


Abbildung 1: Makrorekorder einschalten – Optionen für experimentelle (instabile) Funktionen

In LibreOffice werden die meisten Programmfunktionen durch Absetzen (dispatch) eines Kommandos eingeleitet, welches dann abgefangen und umgesetzt wird. Der Makro-Recorder nimmt genau diese abgesetzten Kommandos auf. Weitergehende Informationen finden Sie im Abschnitt „Dispatch Framework: was ist das?“ auf Seite 14.

## Ein vorhandenes Makro hinzufügen

Der erste Schritt, Makroprogrammierung zu erlernen, ist, vorhandene Makros zu finden und zu benutzen. Im Folgenden gehen wir davon aus, dass Ihnen ein Quelltext eines Makro vorliegt. Es

spielt keine Rolle, ob das Makro aus dem Internet, einer E-Mail oder einem LibreOffice-Dokument stammt. Ein einfaches Makro ist im folgenden Beispiel aufgeführt:

*Beispiel für ein einfaches Makro, das Hallo sagt*

```
Sub HalloWelt
Print "Hallo"
End Sub
```

Um dieses Makro auszuführen, müssen Sie dieses zunächst in einem sogenannten Modul abspeichern. Module sind wiederum in Bibliotheken zusammengefasst. Bibliotheken werden wiederum in Bibliothekscontainern gespeichert. Weitergehende Informationen finden Sie im Abschnitt „Makroorganisation“ auf Seite 15. Zum Erstellen der Bibliothek, die Ihr Makro aufnehmen soll, verfahren Sie wie folgt:

1. Öffnen Sie den Dialog Makro mit **Extras** → **Makros** → **Makros verwalten** → **LibreOffice Basic...** (Abbildung 2)
2. Klicken Sie auf **Verwalten**, um den Dialog *Verwalten* zu öffnen (Abbildung 3).
3. Öffnen Sie die Registerkarte **Bibliotheken**.
4. Wählen Sie den Ort „Meine Makros und Dialoge“ (dieser Ort sollte voreingestellt sein).
5. Öffnen Sie durch einen Klick auf **Neu** den Dialog *Neue Bibliothek*. Geben Sie einen Bibliotheksnamen ein (zum Beispiel „TestBibliothek“) und klicken Sie auf **OK**.
6. Wählen Sie die Registerkarte **Module**.
7. In der Modulliste erweitern Sie den Punkt *Meine Makros* und wählen *TestBibliothek*. Ein Modul namens „Modul1“ existiert bereits und kann Ihr Makro aufnehmen. Sie können aber auch durch Klick auf **Neu** ein neues Modul in *TestBibliothek* erstellen.
8. Wählen Sie „Modul1“ – oder das Modul, das Sie neu erstellt haben – und öffnen Sie durch Klick auf **Bearbeiten** die integrierte Entwicklungsumgebung (IDE).
9. Die IDE ist ein Texteditor für Makros, der Ihnen erlaubt, Makros einzugeben und zu bearbeiten. Kopieren Sie das obige Makro in die IDE (oder tippen Sie es ein).



Abbildung 2: Makros verwalten (LibreOffice Basic)

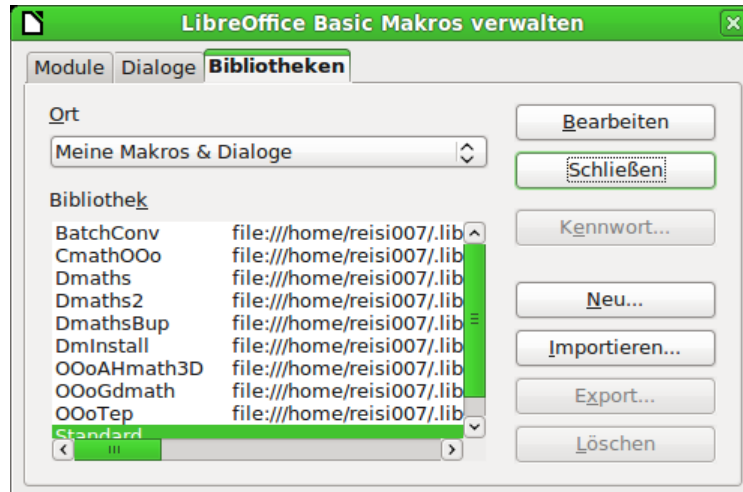


Abbildung 3: LibreOffice Basic Makros verwalten

Beim Anlegen eines neuen Moduls wird automatisch ein leeres funktionsloses Makro namens „Main“ mit folgender Syntax angelegt:

```
Sub Main
End Sub
```

Fügen Sie Ihr neues Makro entweder vor der Zeile „Sub Main“ oder nach der Zeile „End Sub“ ein. Im folgenden Beispiel wurde das neue Makro vor „Sub Main“ eingefügt.

```
REM ***** BASIC *****

Sub HalloWelt
    Print "Hallo"
End Sub

Sub Main
End Sub
```

Drücken Sie die Taste **F5**, um das oberste Makro aus dem Modul auszuführen. Mithilfe des Dialogs *LibreOffice Basic Makros*, der über den Menübefehl **Extras** → **Makros** → **Makros Verwalten** → **LibreOffice Basic...** aufgerufen wird, können Sie auf jedes beliebige Makro in einem Modul zugreifen und es ausführen.

## Ein einfaches Makro selbst erstellen

Für die wiederholte Eingabe von identischen Informationen können Sie die Zwischenablage benutzen. Aber die Zwischenablage wird bei jeder neuen Nutzung verändert, sodass Ihre Information unter Umständen verloren geht. Wenn Sie die Information als Makro speichern, können Sie diese unkompliziert wiederverwenden. In vielen einfachen Fällen – wie auch in unserem Beispiel – können Sie statt eines Makros ebenso auch einen AutoText verwenden. Allerdings soll an dieser Stelle der beispielhafte Einsatz von Makros dargestellt werden.

1. Starten Sie mittels **Extras** → **Makros** → **Makro aufzeichnen...** den Makro-Recorder. Ein kleines Dialogfenster zeigt die laufende Aufnahme an (Abbildung 4).

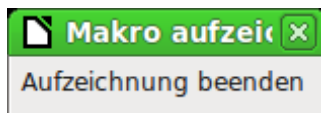


Abbildung 4: Makro aufzeichnen

2. Tippen Sie nun den zu speichernden Text ein oder führen Sie die gewünschten Schritte aus. Im Beispiel wird der Schriftzug „LibreOffice“ benutzt.
3. Klicken Sie auf die Schaltfläche **Aufzeichnung beenden** im Dialog *Makro aufzeichnen*, um die Aufnahme zu beenden, das Makro zu speichern und den Dialog "LibreOffice Basic Makros" zu öffnen (Abbildung 5).

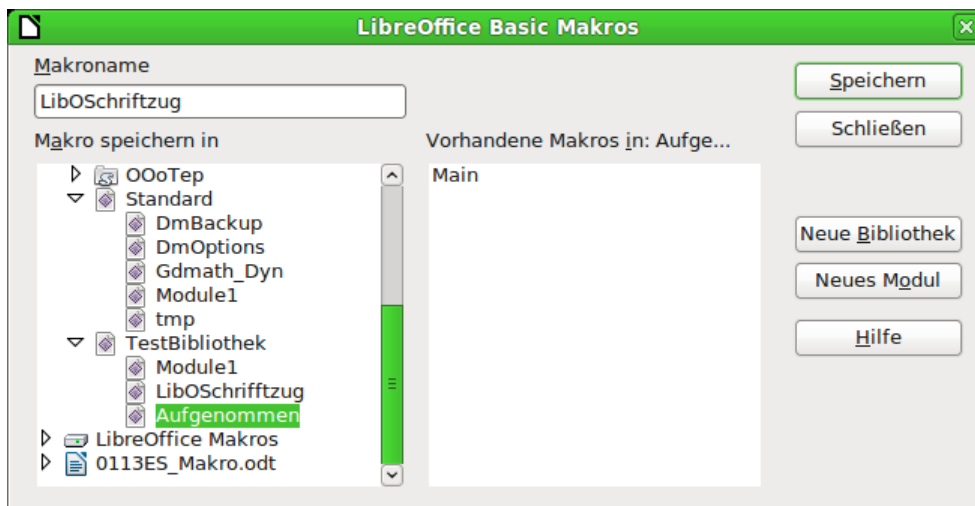


Abbildung 5

4. Öffnen Sie den Bibliothekscontainer „Meine Makros“ und wählen Sie darin die Bibliothek „TestBibliothek“. Bitte beachten Sie, dass alle Bibliotheken ein Modul dieses Namens enthalten. Klicken Sie auf die Schaltfläche **Neues Modul**, um ein neues Modul zu erzeugen, welches das aufgezeichnete Makro aufnehmen soll.
5. Der voreingestellte Modulname lautet „Modul1“. Überschreiben Sie diesen durch einen aussagekräftigeren – zum Beispiel "Aufgenommen" (Vergleichen Sie Abbildung 5).
6. Wählen Sie das erstellte Modul aus und geben Sie in das Feld oben links den gewünschten Namen des Makros ein – zum Beispiel „LibOSchriftzug“. Klicken Sie auf **Speichern**, um das Makro zu speichern.

Wenn Sie alle Schritte wie beschrieben ausgeführt haben, enthält die Bibliothek „TestBibliothek“ nun ein Modul namens „Aufgenommen“. Hier sieht man auch das beim Erstellen des Moduls automatisch hinzugefügte Makro namens „Main“.

## Makro ausführen

Öffnen Sie mittels **Extras** → **Makros** → **Makro ausführen...** den Dialog *Makro-Selektor*. Wählen Sie das oben erstellte Makro aus und klicken Sie auf die Schaltfläche **Ausführen**.



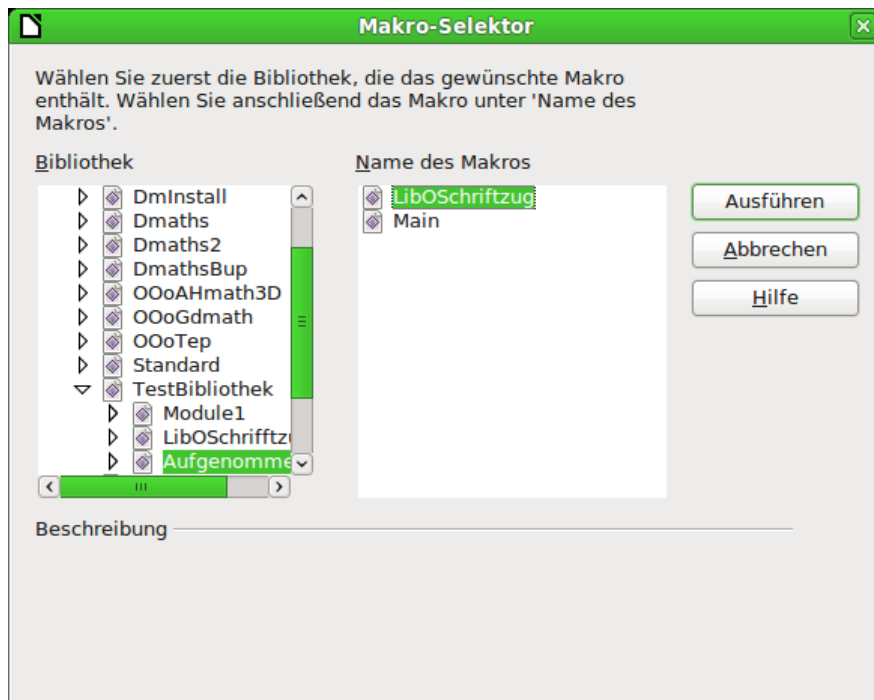


Abbildung 6: Wählen Sie Ihr Makro aus und klicken Sie auf Ausführen.

Makros können auch auf andere Weise ausgeführt werden. So können Sie zum Beispiel über **Extras** → **Makros** → **Makros verwalten** → **LibreOffice Basic...** den Dialog *LibreOffice Basic Makros* aufrufen, der ebenfalls eine Schaltfläche **Ausführen** enthält. Wenn Sie in diesem Dialog auf **Verwalten** klicken, starten Sie den Dialog *Makro Organizer*. Dieser Dialog öffnet gewöhnlich schneller. Allerdings ist die Auswahl etwas langsamer.

## Makro-Code ansehen und bearbeiten

Öffnen Sie den Dialog LibreOffice Basic Makros mittels **Extras** → **Makros** → **Makros verwalten** → **LibreOffice Basic...** Wählen Sie das weiter oben gespeicherte Makro aus und klicken Sie auf Bearbeiten. Die Basic-IDE öffnet sich und das Makro kann hier bearbeitet werden.

*Dieser Makro-Code wurde durch die Aufzeichnung erzeugt*

```
REM ***** BASIC *****

Sub Main

End Sub

Sub LibOSchriftzug
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 argsl(0) as new com.sun.star.beans.PropertyValue
argsl(0).Name = "Text"
```

```
args1(0).Value = "LibreOffice"  
  
dispatcher.executeDispatch(document, ".uno:InsertText", "", 0, args1())  
end sub
```

Das Makro sieht komplizierter aus, als es ist. Die nachfolgenden Erklärungen werden Ihnen helfen, die meisten der verwendeten Befehle der Reihe nach zu verstehen.

### Kommentare starten mit REM

Das Schlüsselwort `REM`, eine Abkürzung des englischen Wortes für remark (Anmerkung) leitet einen Makro-Kommentar ein. Der Rest der Zeile nach dem Wort `REM` wird ignoriert. Alternativ kann auch ein einfaches Anführungszeichen oben (') zur Kennzeichnung von Kommentaren benutzt werden.

#### Tip

LibreOffice Basic-Schlüsselworte kennen keine Groß- und Kleinschreibung – demnach sind `REM`, `rem` und `Rem` gleichwertig. Wenn Sie in der API definierte symbolische Konstanten verwenden, sollten Sie die Groß- und Kleinschreibung jedoch beachten, um auf der sicheren Seite zu sein.

### Subroutinen mit SUB festlegen

Routinen (oder Prozeduren) werden durch das Schlüsselwort `SUB` eingeleitet.

Einzelne Makros werden in Routinen gespeichert, die mit `SUB` beginnen. Das Ende einer Routine wird durch die Zeile `END SUB` gekennzeichnet. Der Code beginnt mit der Definition der Routine namens „Main“, die leer ist und nichts bewirkt. Die nächste Routine, zum Beispiel „LibOSchriftzug“, enthält den generierten Code.

#### Tip

LibreOffice erstellt jedes Mal eine leere Routine namens Main, wenn ein neues Modul erzeugt wird.

Für Interessierte zwei Anmerkungen:

- Sie können Makros auch Werte (Argumente) übergeben. Dies kann jedoch nicht im Rahmen dieser Einführung abgehandelt werden. Bitte informieren Sie sich in der einschlägigen Literatur darüber. Makros, welche durch Aufzeichnung entstanden sind, akzeptieren keine Argumente.
- Funktionen sind spezielle Routinen, die einen Wert zurückgeben. Sie werden durch das Schlüsselwort `FUNCTION` anstatt `SUB` eingeleitet. Aufgezeichnete Makros sind immer vom Typ `SUB`.

### Variablendefinition mittels DIM

Um eine Information für eine spätere Verwendung zu speichern, können Sie diese auf ein Stück Papier schreiben. Auch eine Variable enthält eine Information, die später gelesen oder verändert werden kann. Der `DIM`-Befehl ähnelt einem Stück Papier, das sie auf Ihren Schreibtisch legen, um später darauf eine Information oder eine Nachricht zu vermerken.

Das Makro „LibOSchriftzug“ definiert die Variablen `document` und `dispatcher` vom Typ *Objekt*. Variablen können auch vom Typ *String*, *Integer* oder *Date* sein. Weiter unten wird noch eine weitere Variable, `args1`, als Feld (*array*) von Property-Werten definiert. Variablen vom Typ *array* können mehrere unterschiedliche Werte enthalten, ähnlich wie unterschiedliche Blätter in einem

einigen Ordner zusammengefasst werden können. Die Werte eines Arrays sind aufsteigend nummeriert und beginnen mit 0. In unserem Fall gibt es nur einen Wert, daher bleibt es bei der 0 in den eckigen Klammern.

## Das Makro im Zusammenhang

Vorab: Für das Gesamtverständnis ist es nicht unbedingt erforderlich, alle der im Folgenden beschriebenen Details zu verstehen.

Die erste Zeile definiert den Start des Makro:

```
Sub LibOSchriftzug
```

Dann werden 2 Variablen deklariert:

```
dim document as object
```

```
dim dispatcher as object
```

ThisComponent bezieht sich auf das aktuelle Dokument.

Die Dokumenteigenschaft (Property) CurrentController referenziert einen Dienst, der das Dokument sozusagen „kontrolliert“. Wenn Sie zum Beispiel etwas eingeben, dann „bemerkt“ das der Current Controller. Der Current Controller schickt (dispatch) dann die Änderungen an den Dokumentrahmen (Frame).

Die Property Frame eines Controllers ergibt den Hauptframe für ein Dokument. Daher referenziert die Variable document in der folgenden Zeile den Dokumentrahmen, der die verschickten Befehle erhält.

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

Die meisten Aufgaben in LibreOffice werden durch ein Kommando, welches abgeschickt (dispatched) wird, ausgeführt. LibreOffice enthält einen Dienst namens DispatchHelper, der den größten Teil der Dispatch-Arbeit erledigt, wenn in Makros Dispatch-Befehle verwendet werden. Die Methode CreateUnoService akzeptiert den Namen eines Dienstes (Service) und versucht, eine Instanz dieses Dienstes zu erzeugen. Nach der Ausführung enthält die Variable dispatcher eine Referenz zu einem DispatchHelper.

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

Dann wird ein Feld (Array) von Eigenschaften (Properties) deklariert. Jede Property hat einen Namen und einen Wert, es handelt sich also um ein Namen/Werte-Paar. Das hier erzeugte Feld besitzt eine Property mit Index Null.

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

Dann wird die Property mit dem Namen „Text“ sowie dem Wert „LibreOffice“ versehen:

```
args1(0).Name = "Text"  
args1(0).Value = "LibreOffice"
```

Der DispatchHelper schickt mit dem Kommando uno:InsertText einen Dispatch an den Dokumentrahmen, der in der Variable Dokument gespeichert ist. Die Erklärung der Bedeutung der beiden Argumente „Rahmennummer“ und „Suchflags“ würde diese Einführung sprengen. Das letzte Argument schließlich ist das Feld mit den Property-Werten, die bei der Ausführung des Befehls InsertText benutzt werden sollen.

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

Als letztes wird die Routine beendet:

```
end sub
```

## Makro erstellen

---

Vor Erstellung eines Makros ist es wichtig, sich zwei Fragen zu stellen:

1. Kann die Aufgabe als eine einfache Reihe von Befehlen geschrieben werden?
2. Können die Schritte so angeordnet werden, dass der letzte Befehl den Cursor für den nächsten Befehl an der richtigen Stelle bereitstellt?

## Die integrierte Entwicklungsumgebung

Eine integrierte Entwicklungsumgebung (Integrated development environment, kurz IDE) ist eine Zusammenstellung von Programmierwerkzeugen, um die Erstellung von Software zu vereinfachen. LibreOffice enthält sehr leistungsfähige Werkzeuge, mit denen Sie Ihre Makros ausführen, bearbeiten und Fehler darin finden können. Abbildung 7 zeigt Ihnen die wichtigsten Elemente der LibreOffice-IDE. Der große Bereich in der Mitte, in dem der Makrocode angezeigt wird, ist das Editor-Fenster. Viele Funktionen wie Stopp, Haltepunkt, Einzelschritt und das Beobachtungsfenster dienen als einfacher und effektiver Debugger für Makrocode.

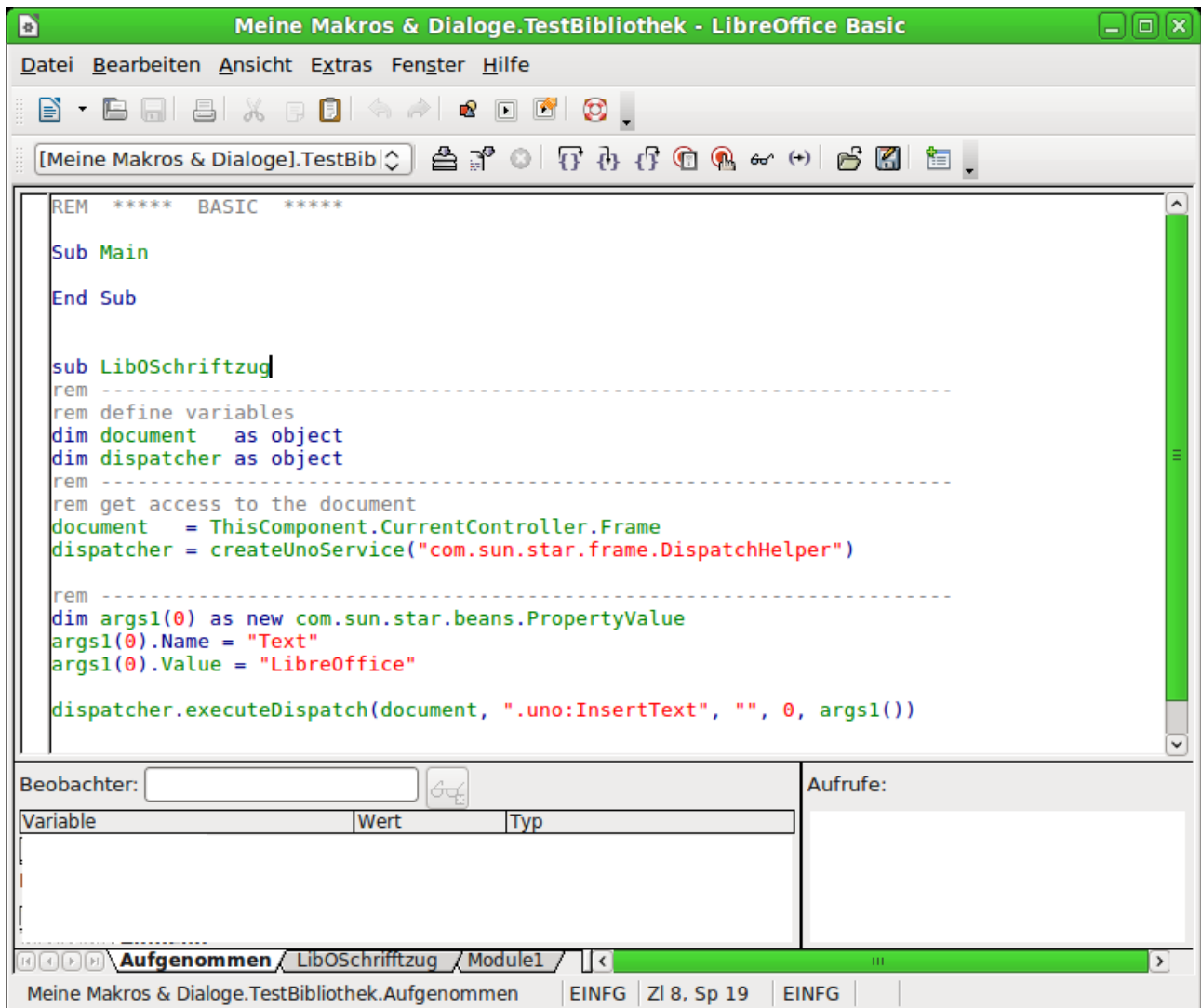


Abbildung 7: IDE

**Hinweis**

Ein Debugger ist einfach ein Werkzeug zum Auffinden, Diagnostizieren und Beheben von Fehlern in Software.

## Schnelles Ausführen eines Makros

---

Wenn Sie ein Makro entwickeln, ist es mühsam, immer wieder das Makro über **Extras** → **Makros** → **Makro ausführen...** laufen zu lassen. Das Makro kann aus der IDE (integrierten Entwicklungsumgebung) ausgeführt werden. Verwenden Sie **Extras** → **Makros** → **Makros verwalten** → **LibreOffice Basic...** und öffnen Sie den Dialog *LibreOffice Basic Makros*.

Wählen Sie das Makro und klicken Sie auf **Bearbeiten**, um das Makro in der IDE zu öffnen. Die IDE hat ein Symbol **Basic-Programm ausführen** in der Symbolleiste, das als erstes Makro in der IDE läuft. Sofern Sie das erste Makro nicht ändern, ist es das leere Makro mit dem Namen **Main**. Ändern Sie **Main**, sodass es folgendermaßen lautet.

```
Sub Main
    CopyNumToCol1
End Sub
```

Jetzt können Sie das Makro "CopyNumToCol1" durch wiederholtes Klicken auf das Symbol **Basic-Programm ausführen** in der Symbolleiste des IDE laufen lassen. Dies ist sehr einfach und schnell – vor allem für temporäre Makros, die ein paar Mal verwendet und dann verworfen werden.

## Bekannte Probleme mit dem Makrorekorder

---

Die Makroaufzeichnung ist leider relativ fehleranfällig. Hauptgrund ist der Dispatch framework bzw. seine Beziehung zum Makrorekorder.

### Dispatch Framework: was ist das?

Der Zweck des Dispatch Framework ist es, einen einheitlichen Zugang zu den LibreOffice-Komponenten für Befehle zu schaffen, die in der Regel den Menüpunkten entsprechen. Man kann **Datei** → **Speichern...** aus dem Menü, die Tastenkombination *Strg* + *S* oder das Symbol **Speichern** in der Symbolleiste *Standard* verwenden, um ein Dokument zu speichern. Alle diese Befehle sind im gleichen "Dispatchbefehl" übersetzt.

Der Dispatch Framework kann auch verwendet werden, um „Befehle“ zurück an das UI (User Interface) zu geben. Zum Beispiel wird nach dem Speichern des Dokuments das Symbol und der Menüeintrag **Datei speichern** deaktiviert. Sobald das Dokument geändert wurde, wird der Eintrag **Speichern** wieder aktiviert.

Beispiele für einen Dispatchbefehl sind `.uno:InsertObject` oder `.uno:GoToStartOfLine`. Der Befehl wird an den Rahmen des Dokuments gesendet und der Rahmen übergeht diesen Befehl, bis ein Objekt gefunden wird, das den Befehl behandeln kann.

## Wie der Makrorekorder den Dispatch Framework verwendet

Der Makrorekorder zeichnet die generierten Meldungen auf. Der Rekorder ist relativ einfach zu implementieren und die gleichen Befehle, die ausgegeben werden, werden für eine spätere Verwendung aufgezeichnet. Das Problem ist, dass nicht alle übertragenen Befehle vollständig sind. Zum Beispiel erzeugt das Einfügen eines Objektes folgenden Code:

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

Es ist nicht möglich, festzulegen, welche Objektart zu erstellen oder einzufügen ist. Wenn ein Objekt aus einer Datei eingefügt werden soll, kann nicht festgelegt werden, welche Datei benutzt werden soll.

Wenn Sie versuchen, ein Makro aufzunehmen und dabei mit **Extras** → **Optionen...** die Optionen zu öffnen und Konfigurationseinstellungen zu ändern, nimmt das erstellte Makro keine Konfigurationsänderungen auf. Der generierte Code ist auskommentiert, so dass das Makro auch nicht korrekt ablaufen kann.

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

Wenn ein Dialog geöffnet ist, wird der Befehl, der den Dialog öffnet, wahrscheinlich auch erzeugt werden. Alle Einstellungen innerhalb des Dialogs werden gewöhnlich nicht aufgezeichnet. Beispiele hierfür sind der Dialog *Makroorganisation*, der Dialog *Sonderzeichen einfügen* und ähnliche Dialogarten. Andere mögliche Probleme mit dem Makrorekorder können das Einfügen einer Formel, die Einrichtung von Anwenderdaten, das Setzen von Calc-Filtern, Aktionen in Datenbankformularen und das Exportieren eines Dokumentes in ein verschlüsseltes PDF sein. Sie werden erst mit dem Ausprobieren herausfinden können, was funktioniert. Die Aktionen aus dem Dialog *Suchen* werden aber zum Beispiel richtig erfasst.

## Weitere Möglichkeiten

Wenn der Makro-Rekorder nicht in der Lage ist, ein bestimmtes Problem zu lösen, ist die übliche Lösung, den Code durch Nutzung der LibreOffice-Objekte zu schreiben. Leider ist die Programmierung für Objekte am Anfang nicht einfach. Es ist normalerweise am besten, mit einfachen Beispielen zu beginnen und diese dann langsam entsprechend Ihren Erfahrungen verästeln zu lassen. Bereits generierte Makros zu lesen und nachzuvollziehen ist ein guter Einstieg.

## Makroorganisation

---

In LibreOffice werden Makros in Module, die Module in Bibliotheken und die Bibliotheken in Containern gruppiert. Eine Bibliothek wird gewöhnlich als Übergruppierung entweder für eine Sammlung von Makros oder für eine ganze Anwendung verwendet. Module werden gewöhnlich in Funktionen wie zum Beispiel Benutzereingaben und Berechnungen unterteilt. Individuelle Makros sind Subroutinen und Funktionen.

- Ein Container enthält ein oder mehrere Bibliotheken und jede Bibliothek ist in einem Container enthalten (siehe Abbildung 8).
- Eine Bibliothek enthält keine oder mehrere Module und jedes Modul ist in einer Bibliothek enthalten.
- Ein Modul enthält keine oder mehrere Makros und jedes Makro ist in einem Modul enthalten.

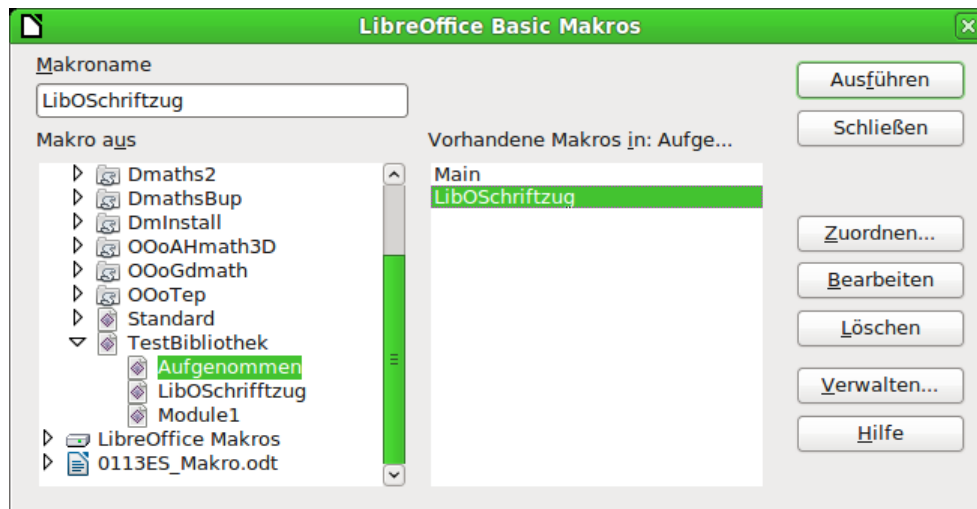


Abbildung 8: Verwaltung von Basic Makros

Verwenden Sie **Extras** → **Makros** → **Makros verwalten** → **LibreOffice Basic...**, um den Dialog *LibreOffice Basic Makros* zu öffnen (Abbildung 8). Es werden alle verfügbaren Bibliothekscontainer in der Liste „Makro aus“ angezeigt. Jedes Dokument ist ein Bibliothekscontainer, der in der Lage ist, mehrere Bibliotheken aufzunehmen. Die Anwendung selbst fungiert als zwei Bibliothekscontainer: ein Bibliothekscontainer für Makros, die mit LibreOffice geliefert werden ("LibreOffice Makros") und ein Bibliothekscontainer für die persönlichen Makros ("Meine Makros"). Die LibreOffice-Makros sind mit dem Runtime-Code der Anwendung gespeichert, die normalerweise nicht bearbeitet werden können, solange Sie nicht als Administrator angemeldet sind. Dies ist auch gut so, da diese Makros nicht geändert werden sollen. Sie sollten auch Ihre eigenen Makros nicht im LibreOffice-Container speichern.

Solange Sie Ihre Makros für ein einzelnes Dokument, und nur in einem einzigen Dokument, benutzbar machen, werden Ihre Makros am sinnvollsten im Container „Meine Makros“ gelagert. Der Container "Meine Makros" wird in Ihrem Benutzerbereich oder Verzeichnis *home* gespeichert.

Wenn ein Makro in einem Dokument enthalten ist, wird ein aufgezeichnetes Makro versuchen, in diesem Dokument zu arbeiten – vor allem wenn es "ThisComponent" für seine Aktionen benutzt.

Jeder Bibliothekscontainer enthält eine Bibliothek namens *Standard*. Es ist besser, eigene Bibliotheken mit aussagekräftigen Namen als die Standard-Bibliothek zu verwenden. Diese Namen sind nicht nur einfacher zu verwalten, sondern Sie können diese auch in andere Bibliothekscontainer importieren, während die Standard-Bibliothek nicht importiert werden kann.

### Vorsicht



LibreOffice erlaubt es Ihnen, Bibliotheken in einem Bibliothekscontainer zu importieren. Aber es ist nicht möglich, dass Sie die Bibliothek namens „Standard“ überschreiben. Wenn Sie Ihre Makros in der Standard-Bibliothek speichern, können diese nicht in einen anderen Bibliothekscontainer importiert werden.

Genauso ist es ratsam, aussagekräftige Namen für Ihre Module zu verwenden. Standardmäßig verwendet LibreOffice Namen wie „Module1“. Es steht Ihnen frei, Ihre eigenen, aussagekräftigen Namen zu verwenden.

Wenn Sie Makros erstellen, müssen Sie auch entscheiden, wo diese abgespeichert werden sollen. Das Abspeichern eines Makros in einem Dokument ist dann nützlich, wenn das Dokument weiter gegeben wird und Sie wollen, dass das Makro im Dokument enthalten ist. Makros, die im



Anwendungsbibliothekscontainer namens "Meine Makros" gespeichert sind, sind dagegen global für alle Dokumente verfügbar.

Makros sind erst dann verfügbar, wenn die Bibliothek, die sie enthält, geladen ist. Die Standard-Bibliothek und Template-Bibliothek werden jedoch automatisch geladen. Eine geladene Bibliothek wird anders als eine Bibliothek, die nicht geladen ist, angezeigt. Um die Bibliothek und die enthaltenen Module zu laden, klicken Sie doppelt auf die Bibliothek.

## Wo werden Makros gespeichert?

LibreOffice speichert benutzerspezifische Daten in ein Verzeichnis unter dem Benutzerverzeichnis. Die Lage des Benutzerverzeichnisses ist abhängig von dem verwendeten Betriebssystem. Verwenden Sie **Extras** → **Optionen...** → **LibreOffice** → **Pfade**, um zu sehen, wo andere Konfigurationsdaten gespeichert werden. User-Makros sind in *LibreOffice\3\user\basic* gespeichert. Jede Bibliothek ist in einem eigenen Verzeichnis unterhalb des Basisverzeichnisses gespeichert.

Es ist für den gelegentlichen Gebrauch nicht wichtig zu verstehen, wo Makros gespeichert werden. Wenn Sie aber wissen, wo sie gespeichert sind, können Sie eine Sicherungskopie erstellen, Ihre Makros weitergeben oder überprüfen, ob ein Fehler vorhanden ist. Gerade wenn ein Makro einen Fehler enthält, können Sie das Makro jemandem schicken, der Ihnen eventuell bei der Fehlersuche helfen kann.

Verwenden Sie **Extras** → **Makros** → **Dialoge verwalten...**, um den Dialog *LibreOffice Makros verwalten* zu öffnen. Eine weitere gängige Methode, um diesen Dialog zu öffnen, ist über **Extras** → **Makros** → **Makros verwalten** → **LibreOffice Basic...** das Dialogfeld *Makros* zu öffnen, auf die Schaltfläche **Verwalten** zu klicken und zum Reiter *Dialoge* zu wechseln.



Abbildung 9: Reiter *Dialoge* im Fenster *LibreOffice Basic Makros verwalten*

## Makros importieren

Der Dialog *Makro verwalten* bietet Funktionen zum Erstellen, Löschen und Umbenennen von Bibliotheken, Modulen und Dialogen. Wählen Sie den Container *Bibliothek* und klicken Sie dann auf die Schaltfläche **Importieren**, um Makro-Bibliotheken zu importieren (Abbildung 10).

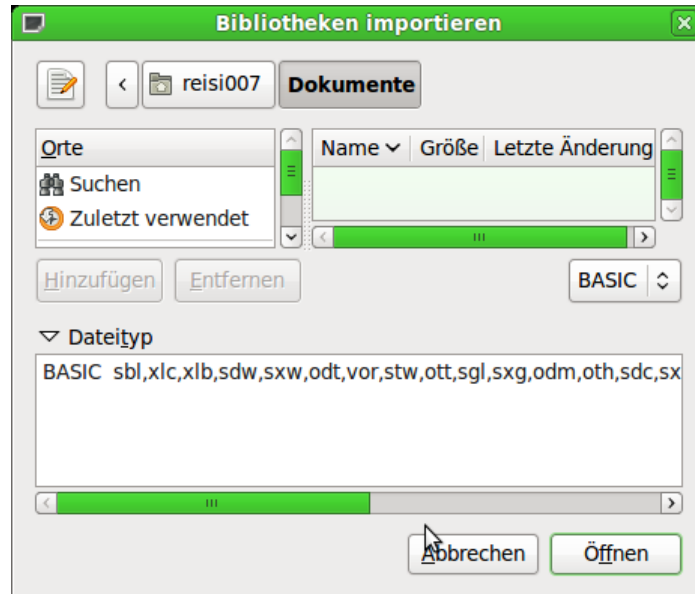


Abbildung 10: Bibliothek importieren

### Hinweis

Eine Bibliothek „Standard“ kann nicht importiert werden.

### Tip

Unter Linux werden die LibreOffice-spezifischen Dateien unter dem Benutzer-Home-Verzeichnis in einem Unterverzeichnis gespeichert, dessen Name mit einem Punkt beginnt. Verzeichnisse und Dateien, die mit einem Namen mit einem Punkt beginnen, werden in einem normalen Auswahldialog nicht angezeigt. Um das Verzeichnis zu öffnen, gehen Sie entweder zum Home-Verzeichnis und geben den Namen `.LibreOffice\3` ein und klicken dann auf **Öffnen** oder Sie drücken *Strg-H*, um versteckte Dateien und Ordner anzuzeigen und navigieren wie gewohnt. Es öffnet sich das Verzeichnis, das nicht von Anfang an gezeigt wurde.

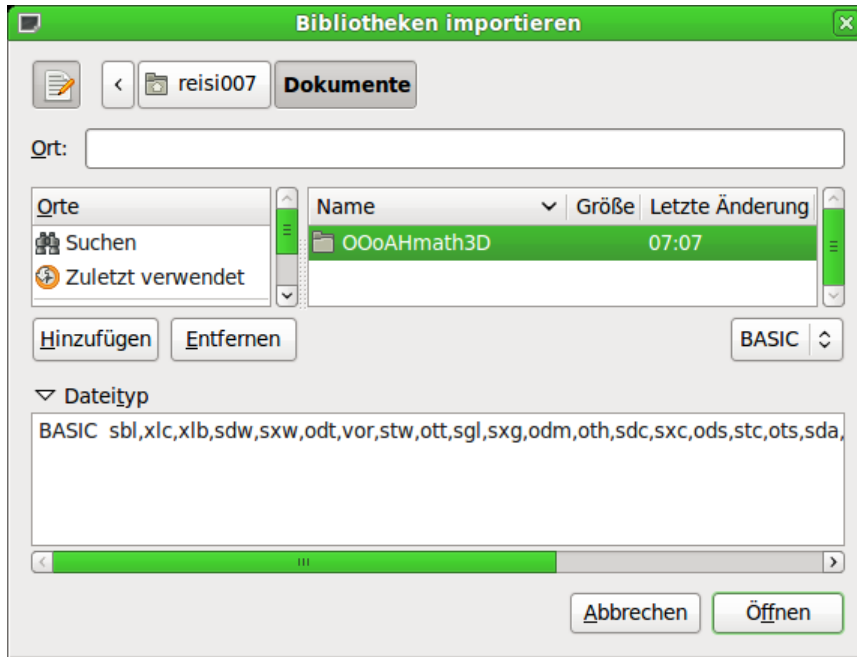


Abbildung 11: Bibliothek importieren 1

Gehen Sie zu dem Verzeichnis, das die Bibliothek enthält, die Sie importieren möchten. Normalerweise gibt es zwei Dateien zur Auswahl: dialog.xlb und script.xlb. Es spielt keine Rolle, welche der beiden Dateien Sie wählen, da beide Dateien importiert werden. Wählen Sie eine Datei und klicken Sie auf **Öffnen**, um fortzufahren.

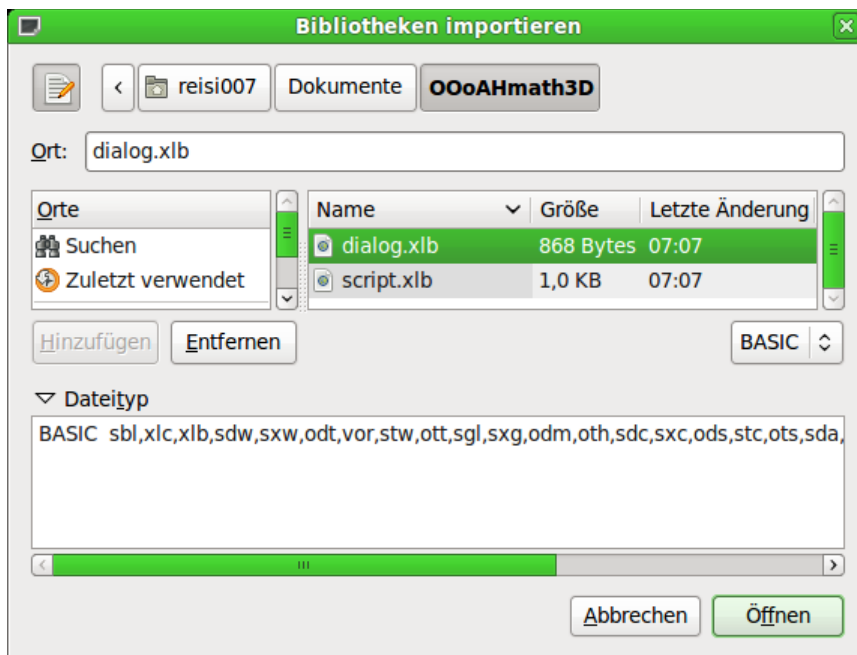


Abbildung 12: Bibliothek importieren 2

Wenn die Bibliothek bereits vorhanden ist, wird sie nicht ersetzt, es sei denn „Existierende Bibliotheken ersetzen“ ist markiert. Wenn „Als Referenz einfügen (nur lesen)“ aktiviert ist, wird die Bibliothek an ihrem aktuellen Speicherort referenziert, aber Sie können diese nicht bearbeiten. Wenn „Als Referenz einfügen (nur lesen)“ nicht aktiviert ist, wird die Bibliothek in das Makroverzeichnis des Benutzers kopiert.

Makros können in Bibliotheken innerhalb von LibreOffice-Dokumenten gespeichert werden. Wählen Sie ein Dokument anstelle eines Verzeichnis auf der Festplatte (wie in Abbildung 13 gezeigt), um die in einem Dokument enthaltenen Bibliothek zu importieren.

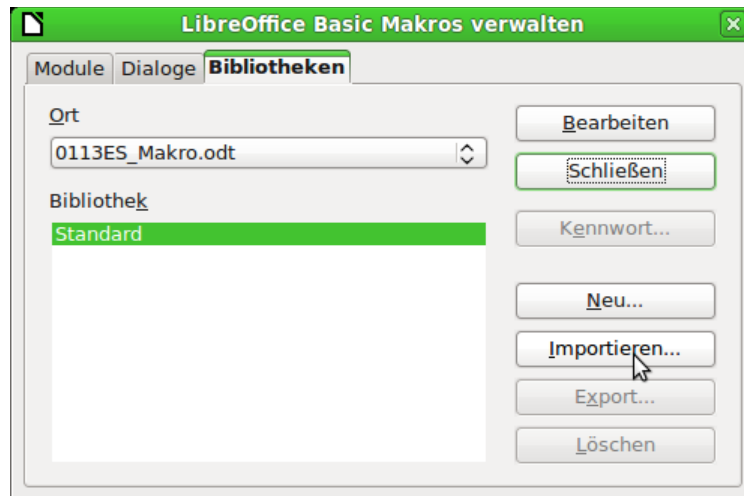


Abbildung 13: Makros in Dokument importieren

## Makros für den Import herunterladen

Makros stehen auch zum Herunterladen zur Verfügung. Einige Makros werden in Dokumenten, manche als reguläre Dateien, die Sie auswählen und importieren müssen, und einige als Makrotext bereit gestellt, der in die Basic IDE kopiert und eingefügt werden soll. Verwenden Sie **Extras** → **Makros** → **Makros verwalten** → **LibreOffice Basic...**, um den Dialog *LibreOffice Basic Makros* zu öffnen. Wählen Sie das Makro, um es zu bearbeiten, und klicken Sie dann auf **Bearbeiten**, um das Makro in der Basic IDE zu öffnen.

Einige Makros finden sich auch als kostenlose Downloads im Internet (Vergleichen Sie Tabelle 1). „Ein vorhandenes Makro hinzufügen“ auf Seite 5 beschreibt, wie Sie diese Makros Ihrer Makrobibliothek hinzuzufügen.

Tabelle 1: Stellen, an denen man Makrobeispiele findet (meist englischsprachig)

Ort	Beschreibung
<a href="http://wiki.documentfoundation.org/Macros/de">http://wiki.documentfoundation.org/Macros/de</a>	Wiki der The Document Foundation speziell zum Thema „Makros“
<a href="http://www.oocomacros.org/">http://www.oocomacros.org/</a>	Sammlung von Makros aus dem OOo-Projekt.
<a href="http://www.pitonyak.org/oo.php">http://www.pitonyak.org/oo.php</a>	Referenzmaterialien über Makros.
<a href="http://development.openoffice.org/">http://development.openoffice.org/</a>	Viele Links zu allem.
<a href="http://www.ooforum.org/">http://www.ooforum.org/</a>	Ein Forum, mit vielen Beispielen und Hilfe.
<a href="http://user.services.openoffice.org/">http://user.services.openoffice.org/</a>	Ein weiteres Forum, mit vielen Beispielen und Hilfe.

## Wie man ein Makro ausführt

Eine typische Methode, um ein Makro auszuführen, ist im Folgenden beschrieben:

1. Verwenden Sie **Extras** → **Makros** → **Makro ausführen...**, um den Dialog Makro Selektor zu öffnen (Abbildung 14).
2. Wählen Sie die Bibliothek und das Modul in der Bibliothek-Liste (linke Seite).
3. Wählen Sie das Makro in der Liste *Name des Makros* (rechte Seite).
4. Klicken Sie auf **Ausführen**, um das Makro auszuführen.

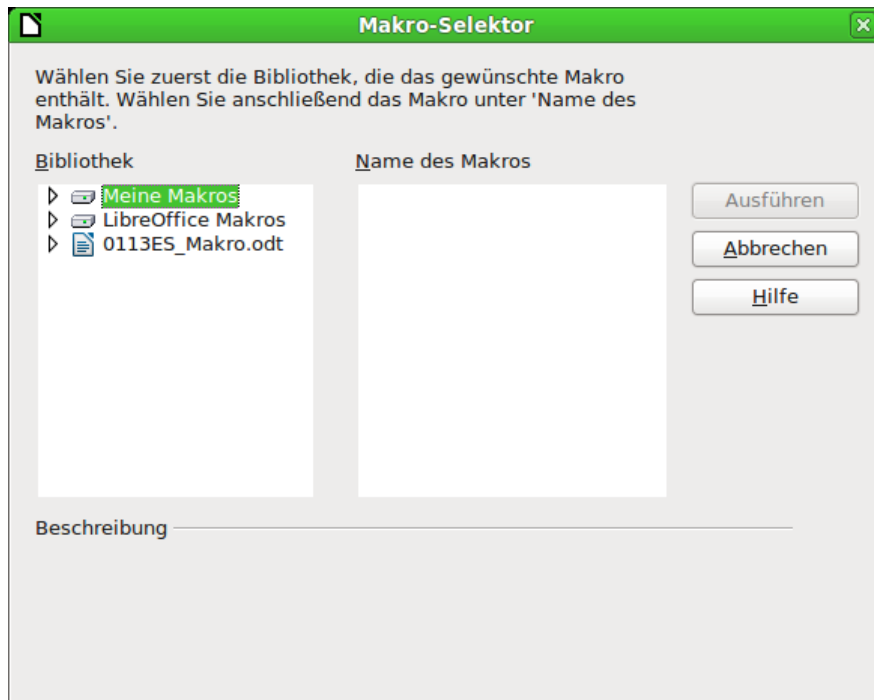


Abbildung 14: Makro Selektor

Obwohl Sie **Extras** → **Makros** → **Makro ausführen...** jedes Mal benutzen können, um Makros auszuführen, ist dies keine effiziente Vorgehensweise für häufig genutzte Makros. Sinnvoller ist es hierfür, ein Makro einer Symbolleiste, einem Menüpunkt, einem Tastaturkürzel oder einer Schaltfläche, die in einem Dokument eingebettet ist, zuzuweisen. Bei der Auswahl einer Methode ist es auch gut, sich folgende Fragen zu stellen:

- Soll das Makro nur in einem Dokument zur Verfügung stehen oder global für alle Dokumente?
- Bezieht sich das Makro auf einen bestimmten Dokumenttyp, wie z. B. ein Calc-Dokument?
- Wie häufig wird das Makro verwendet werden?

Die Antworten werden bestimmen, wo das Makro gespeichert und wie es zugänglich gemacht wird. Sie werden zum Beispiel ein selten verwendetes Makro wahrscheinlich nicht in einer Symbolleiste einfügen. Um Sie bei Ihrer Wahl zu unterstützen haben wir die folgende Tabelle 2 erstellt:

Tabelle 2: Methoden, um ein Makro zu starten

Typ	LibreOffice	Dokumenttyp	Dokument
Werkzeugleiste	Nein	Ja	Ja
Menü	Nein	Ja	Ja
Kurztaste	Ja	Ja	Nein
Ereignis	Ja	Nein	Ja

Um einen Menüpunkt, Tastenkombination oder ein Symbol in der Symbolleiste für den Makroaufruf einzufügen, verwenden Sie das Dialogfenster Anpassen (Abbildung 15). Öffnen Sie diesen Dialog mit einer der folgenden Arten:

- Wählen Sie **Extras** → **Anpassen...** aus der Menüleiste.
- Jede Symbolleiste hat ein Symbol (auf der Spitze stehendes Dreieck), das ein Menü öffnet. Wählen Sie **Symbolleiste anpassen**.

Der Dialog *Anpassen* enthält die Register *Menüs*, *Tastatur*, *Symbolleisten* und *Ereignisse*. Weitere Erklärungen bezüglich des Dialogs finden Sie im Kapitel 14 „Anpassen von LibreOffice“ dieses Handbuchs.

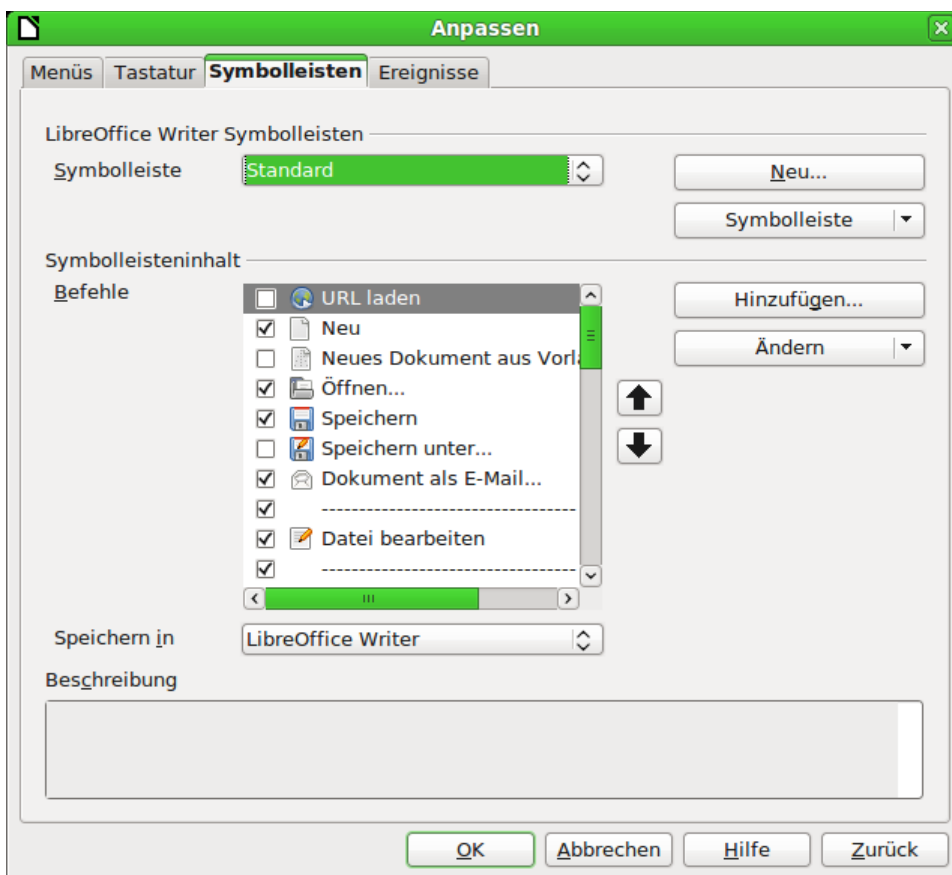


Abbildung 15: Symbolleisten anpassen

## Symboleiste

Makros können der Symboleiste hinzugefügt werden. Einzelheiten hierzu finden Sie im Kapitel 14 „Anpassen von LibreOffice“ dieses Handbuchs.

## Menüeintrag

Über **Extras** → **Anpassen...** wird der Dialog *Anpassen* geöffnet. Dort wählen Sie die Registerkarte **Menü**. Sie können dann einen vorhandenen Menüeintrag bearbeiten oder neue Menüeinträge zum Aufruf von Makros erstellen. Weitere Informationen zur Menüanpassung finden Sie im Kapitel 14 „Anpassen von LibreOffice“ dieses Handbuchs.

## Tastaturkürzel

Über **Extras** → **Anpassen...** wird der Dialog *Anpassen* geöffnet, dort wählen Sie die Registerkarte **Tastatur**. Einzelheiten zur Zuweisung von Tastaturkürzeln finden Sie im Kapitel 14 „Anpassen von LibreOffice“ dieses Handbuchs.

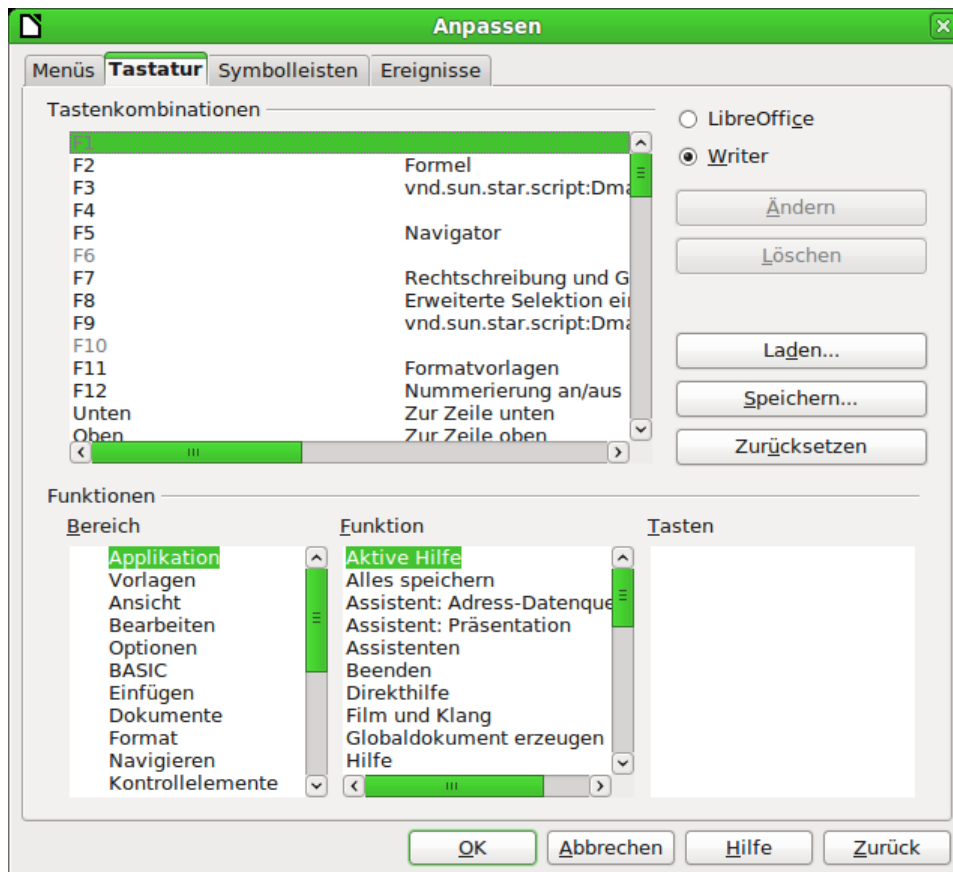


Abbildung 16: Anpassen -- Tastatur

## Ereignis

Wenn in LibreOffice etwas passiert, spricht man von einem Ereignis oder englisch Event. Ereignisse treten zum Beispiel auf, wenn ein Dokument geöffnet, eine Taste gedrückt oder die Maus bewegt wird. LibreOffice ermöglicht es, Ereignissen ein Makro zuzuordnen, wenn das Ereignis eintritt. Das Makro wird dann Event-Handler genannt. Die vollständige Beschreibung von

Event-Handler geht weit über den Rahmen dieses Handbuches hinaus. Hier soll nur ein kleiner Einblick gegeben werden.

### Vorsicht



Seien Sie vorsichtig, wenn Sie einen Event-Handler konfigurieren. Wenn Sie einmal angenommen einen Event-Handler für jeden Tastendruck konfigurieren und dabei einen Fehler machen, sodass das Ereignis nicht richtig gehandhabt wird, kann es sein, dass das Programm bei jedem Tastendruck nicht mehr reagiert und Sie LibreOffice beenden müssen.

Verwenden Sie **Extras** → **Anpassen...**, um den Dialog *Anpassen* zu öffnen, und wählen Sie die Registerkarte **Ereignisse** (Abbildung 17). Die Ereignisse im Dialog *Anpassen* beziehen sich auf die gesamte Anwendung und bestimmte Dokumente. Verwenden Sie das Feld *Speichern in*, um LibreOffice oder ein bestimmtes Dokument zu wählen.

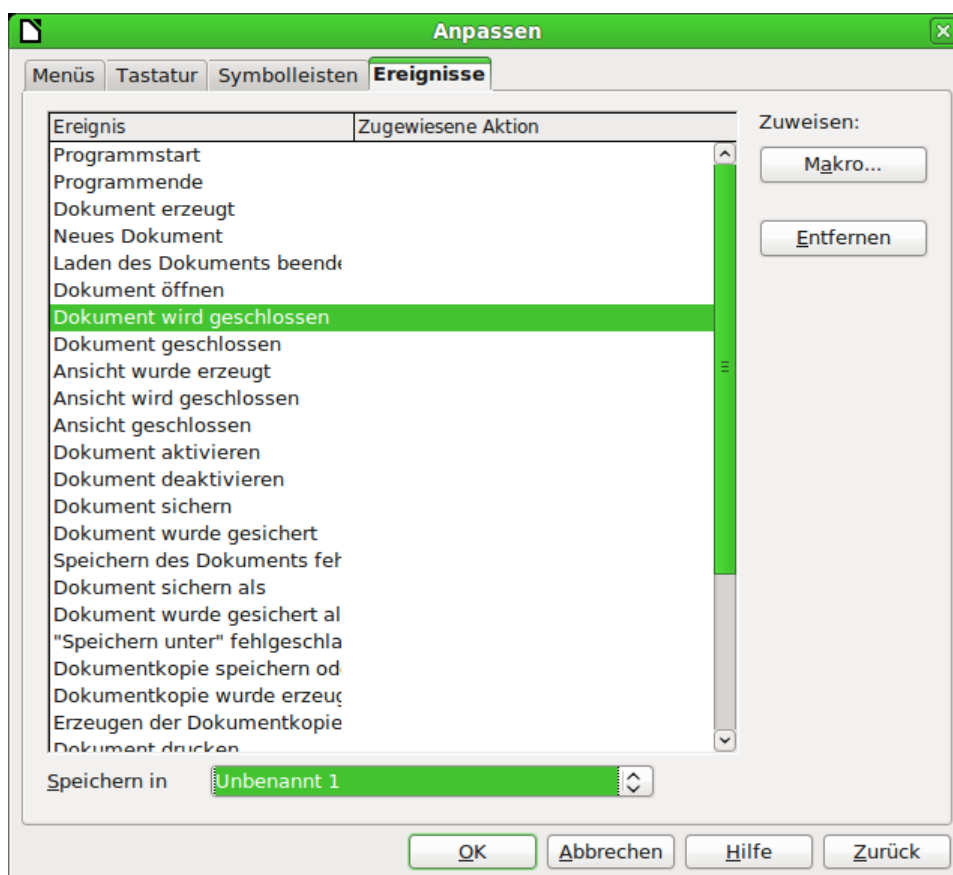


Abbildung 17: *Anpassen – Ereignisse I*

Man kann zum Beispiel dem Ereignis „Dokument öffnen“ das Starten eines bestimmten Makros zuweisen. Das Makro führt dann bestimmte Setup-Aufgaben für das Dokument durch. Wählen Sie das gewünschte Ereignis und klicken Sie auf die Makrotaste, um den Makro-Auswahldialog zu öffnen (Abbildung 17).

Wählen Sie das gewünschte Makro und klicken Sie auf **OK**, um das Makro dem Ereignis zuzuordnen. Die Registerkarte Ereignisse zeigt, dass ein Makro einem Ereignis zugeordnet wurde (Abbildung 18). Wenn in dem aufgeführten Beispiel ein Dokument geöffnet wird, wird das PrintHello-Makro ausgeführt.

Viele Objekte in einem Dokument können dazu verwendet werden, ein Makro aufzurufen, wenn ein bestimmtes Ereignis eintritt. Die häufigste Verwendung ist ein Steuerelement einzufügen, z. B. eine



Schaltfläche in einem Dokument. Bei einem Doppelklick auf eine Grafik öffnet sich der Dialog *Bild* mit einer Registerkarte *Makro*, sodass Sie einem Ereignis ein Makro zuweisen können.

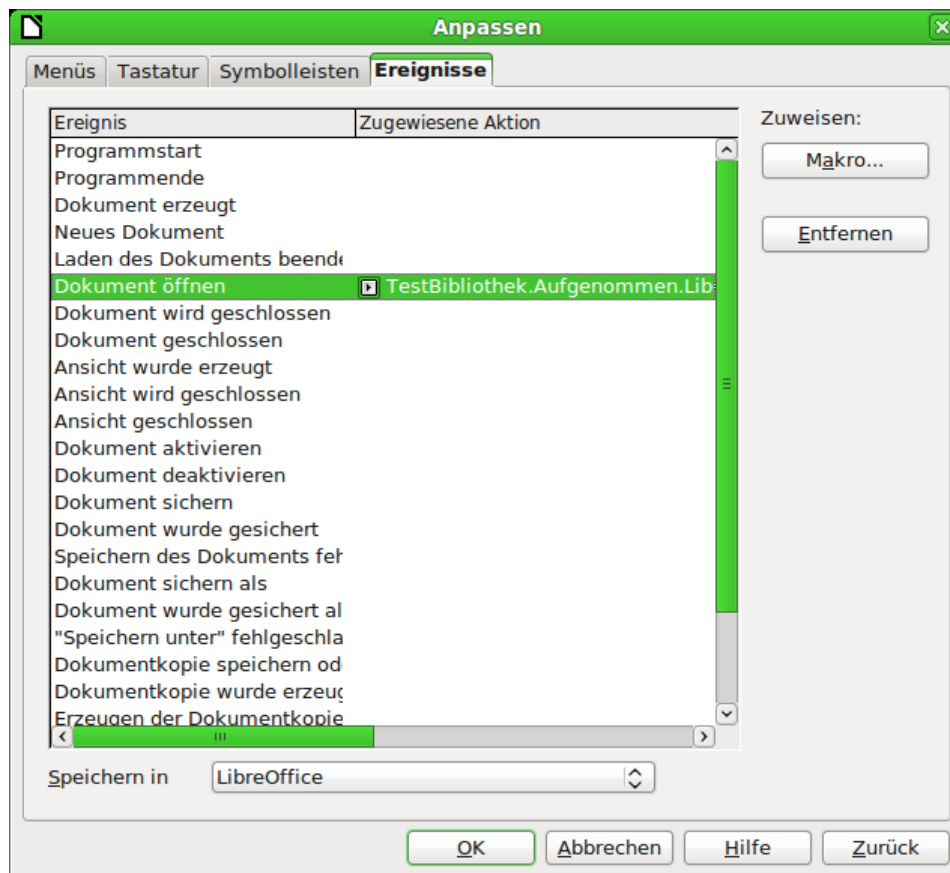


Abbildung 18: Anpassen -- Ereignisse II

## Erweiterungen (Extensions)

Eine Erweiterung (Extension) ist ein Paket, das in LibreOffice integriert werden kann, um bestimmte Aufgaben zu erledigen. Insofern stellt eine Erweiterung so etwas Ähnliches wie ein Makro dar. Der Unterschied ist, dass ein Makro eine gespeicherte Abfolge von Kommandos oder Tastatureingaben ist, die ein Anwender selbst erstellen kann. Eine Erweiterung ist auch eine Abfolge von Kommandos oder Tastatureingaben. Allerdings kann der Anwender den Quellcode nicht ändern.

An dieser Stelle soll explizit darauf hingewiesen werden, dass es sinnvoll ist, im Internet nach Erweiterungen zu suchen, die eventuell die Fragestellung lösen können, ohne dass ein Makro erstellt werden muss.

Erweiterungen können in folgende Gruppen unterteilt werden:

- Neue Komponenten und Funktionen, die normalerweise auch ein gewisses Maß an Userinterface-Integration wie neue Menüs oder Symbolleisten bieten.
- Calc Add-Ins, die neue Funktionalitäten für Calc bieten, einschließlich neuer Funktionen, die wie normale eingebaute Funktionen zur Verfügung stehen.
- Daten-Piloten, die direkt in Calc verwendet werden.
- Diagramm-Add-Ins mit neuen Diagrammtypen.

- Linguistische Komponenten wie Rechtschreib-, Stil- und Grammatikprüfung.
- Dokumentvorlagen und Bilder.

Erweiterungen können in fast jeder Programmiersprache geschrieben werden und können einfach oder anspruchsvoll sein. Weitere Informationen, wie Sie Erweiterungen erhalten und installieren, finden Sie in Kapitel 14 „Anpassen von LibreOffice“ dieses Handbuchs.

## Makros schreiben ohne Recorder

---

Die Beispiele, die in diesem Kapitel gezeigt werden, wurden mit dem Makrorekorder oder dem Dispatcher erstellt. Sie können auch Makros schreiben, die direkt auf die Objekte zugreifen, die in LibreOffice enthalten sind. Mit anderen Worten, Sie können ein Dokument direkt beeinflussen.

Die internen Objekte von LibreOffice direkt zu beeinflussen ist ein Thema für Fortgeschrittene, das über den Rahmen dieses Kapitels hinausgeht. Ein einfaches Beispiel zeigt allerdings, wie dies grundsätzlich funktioniert:

*Beispiel für ein Makro, das den Text "Hallo" an das Ende des Dokuments hängt*

```
Sub HalloalsAnhang
    Dim oDoc
    Dim sTextService$
    Dim oCurs

    REM Diese Komponente betrifft das aktuelle Dokument.
    oDoc = ThisComponent
    REM Überprüfen, ob dies ein Textdokument ist
    sTextService = "com.sun.star.text.TextDocument"
    If NOT oDoc.supportsService(sTextService) Then
        MsgBox "Dieses Makro arbeitet nur mit einem Textdokument"
    Exit Sub
    End If

    REM Abfrage des Zeigers von der aktuellen Steuerung.
    oCurs = oDoc.currentController.getViewCursor ()

    REM Bewege den Mauszeiger zum Ende des Dokuments.
    oCurs.gotoEnd (False)

    REM Fügt den Text "Hallo" am Ende des Dokuments ein
    oCurs.Text.insertString(oCurs, "Hallo", False)
End Sub
```

## Weitere Informationen

---

Es stehen zahlreiche Quellen zur Verfügung, die Hilfe bei der Erstellung von Makros bieten. Verwenden Sie zuerst **Hilfe** → **LibreOffice Hilfe**, um die LibreOffice-Hilfeseiten zu öffnen. Die linke obere Ecke des LibreOffice Hilfesystem enthält eine Auswahlliste. Um die Hilfe für Basic zu erhalten, wählen Sie LibreOffice Basic aus dieser Liste. Dieser Hinweis gilt nur, wenn Sie das Helpack für LibreOffice installiert haben, sodass die Hilfe auch offline zur Verfügung steht.

## In LibreOffice enthaltene Makros

Viele gute Makros sind schon in LibreOffice enthalten. Verwenden Sie **Extras** → **Makros** → **Makros verwalten** → **LibreOffice Basic...**, um den *LibreOffice Basic Makros* zu öffnen. Erweitern

Sie die Bibliothek LibreOfficeMakros. Sehen Sie sich zum Beispiel unter Tools die Debug-Module an. Einige gute Beispiele sind WritedbglInfo (Dokument) und printdbglInfo (Datenblatt). Versuchen Sie, diese Schritt für Schritt nachzuvollziehen.

## Onlinequellen

Die folgenden Links und Referenzen beinhalten Informationen betreffs Makroprogrammierung:

<http://wiki.documentfoundation.org/Macros/de> (Wiki der The Document Foundation speziell zum Thema „Makros“)

<http://api.libreoffice.org/docs/common/ref/com/sun/star/module-ix.html> (Offizielle IDL Referenz, hier finden Sie fast alle Befehle mit einer Beschreibung)

<http://api.openoffice.org/docs/common/ref/com/sun/star/module-ix.html> (Offizielle IDL Referenz von OOO, hier finden Sie fast alle Befehle mit einer Beschreibung)

<http://download.oracle.com/docs/cd/E19535-01/819-0439/> (Das Handbuch von SUN/Oracle zur Programmierung mit Basic für StarOffice 8, sehr gut geschrieben und dargestellt und Grundlage einiger OOO-Programmierhandbücher.)

<http://www.pitonyak.org/oo.php> (Andrew Pitonyaks Makroseite)

---

### Hinweis

Aufgrund der Entwicklung bei OpenOffice.org können wir nicht garantieren, dass die darauf bezogenen Links auch in Zukunft in dieser Form zur Verfügung stehen.

---