

# 1 OpenOffice mit Delphi

In der Ausgabe 3'04 der Toolbox wurde demonstriert, wie MS Word aus Delphi angesprochen wird. In dem Bericht wird die gleiche Funktionalität implementiert, aber dabei OpenOffice.org Writer als Textverarbeitung eingesetzt und OpenOffice.org Calc zum Erzeugen von Spreadsheets aus einem TDataset von Delphi. OpenOffice ist die Open-Source-Implementierung einer Office-Suite, die die gleiche oder ähnliche Funktionalität wie MS Office bietet. Wenn Implementierungs- und Deploymentkosten einer Software eine Rolle spielen, sollte man sich dieses Paket unbedingt ansehen. Zudem gab es einen Bericht OpenOffice im Entwickler 6/2003 unter dem Titel OO-Automaten, der einen grundlegenden Einstieg zeigt, so daß auch via COM ein Weg besteht.

Dieser Bericht ist eher eine Zusammenstellung aus verschiedenen Quellen und soll eine Starthilfe und Einstieg zu obigem Thema bieten.

## 1.1 Sourcen zu OpenOffice mit Delphi

## 1.2 Delphi OOo

*Author: Bernard Marcellly*

This is a toolbox to help programmers drive the application OpenOffice.org from the programming language Delphi using COM Automation technology. It contains:

- A unit with routines similar to some OpenOffice.org Basic routines used to access API,
- A unit which defines all API constants and enumerations
- A Delphi version of the Xray tool
- Typical programming examples and all source code
- A HowTo describing the toolbox

The source code included in the package is released under the LGPL. The instruction document is released under the [Public Documentation License version 1.0](#)

Latest release: Version 1.2 (August 1, 2007)

[http://sourceforge.net/project/showfiles.php?group\\_id=87718&package\\_id=133853](http://sourceforge.net/project/showfiles.php?group_id=87718&package_id=133853)

## 1.3 PDF Generator to OO

In diesem Code Snippet wird auch grundsätzlich gezeigt, wie mit OLE respektive COM oder einer OLE Bridge ein Service Kanal zum OpenOffice Universum geöffnet wird.

I have this Delphi object that I use to successfully convert files (like .DOC and .SWX) to PDF files using OOo. Unfortunately I've recently found that HTML (.HTM) files throw an exception and I'm hoping that someone else here has seen this behavior before and can tell me what I'm doing wrong or if it's a known issue with OOWriter. Eine kleine Klasse zum einfachsten Handling von OpenOffice Dokumenten, diese Funktionen funktionieren auch mit Word .dot Files, das ersetzen als Bookmark geht sogar mit OpenOffice schneller als mit Word selbst. Here is the code:

```
*** Interface ***  
type
```

```

TOOoWriter = class(TObject)
private
  fOpenOffice: Variant;
  fDocument: Variant;
  fConnected: boolean;
  fDocumentOpened: boolean;
  fDesktop: Variant;
  function MakePropertyValue (PropName:string;PropValue:variant):variant;
Public
  Constructor Create;
  destructor Destroy; override;
  function Connect: boolean;
  procedure Disconnect;
  function OpenDocument (Filename:string):boolean;
  procedure SaveToPDF (FileName:string);
  procedure CloseDocument;
end;

*** Implementation ***
{ TOOoWriter }
procedure TOOoWriter.CloseDocument;
begin
  if fDocumentOpened then begin
    fDocument.Close(false);
    fDocumentOpened:= false;
    fDocument:= Unassigned;
    fDesktop.Terminate;
    fDesktop:= UnAssigned;
  end;
end;

function TOOoWriter.Connect: boolean;
begin
  if VarIsEmpty(fOpenOffice) then
    fOpenOffice:= CreateOleObject('com.sun.star.ServiceManager');
  fConnected:= not (VarIsEmpty(fOpenOffice) or VarIsNull(fOpenOffice));
  Result:= fConnected;
end;

constructor TOOoWriter.Create;
begin
  inherited;
  CoInitialize(nil);
end;

destructor TOOoWriter.Destroy;
begin
  CoUninitialize;
  inherited;
end;

procedure TOOoWriter.Disconnect;
begin
  if fDocumentOpened then
    CloseDocument;
  fConnected:= false;

```

```

    fOpenOffice:= Unassigned;
end;

function TOOoWriter.MakePropertyValue(PropName: string;
    PropValue: variant): variant;
var
    Struct: variant;
begin
    Struct:= fOpenOffice.Bridge GetStruct('com.sun.star.beans.PropertyValue');
    Struct.Name:= PropName;
    Struct.Value:= PropValue;
    Result:= Struct;
end;

function TOOoWriter.OpenDocument(FileName: string): boolean;
var
    wProperties: Variant;
begin
    if not fConnected then abort;
    fDesktop:= fOpenOffice.createInstance('com.sun.star.frame.Desktop');
    wProperties:= VarArrayCreate([0, 0], varVariant);
    wProperties[0]:= MakePropertyValue('Hidden', True);
    fDocument:= fDesktop.loadComponentFromURL('file:///'+ StripString(FileName,
['\'], '/') , 'blank', 0, wProperties);
    fDocumentOpened:= not (VarIsEmpty(fDocument) or VarIsNull(fDocument));
    result:= fDocumentOpened;
end;

procedure TOOoWriter.SaveToPDF(FileName: string);
var
    wProperties: variant;
begin
    if not (fConnected and fDocumentOpened) then abort;
    wProperties:= VarArrayCreate([0, 0], varVariant);
    wProperties[0]:= MakePropertyValue('FilterName', 'writer pdf Export');
    fDocument.StoreToURL('file:///'+ StripString(FileName, ['\'], '/'), wProperties);
end;

```

Ein interessanter und effektiver aber eher ungewöhnlicher Ansatz ist es das Programm mit Tastaturkürzeln fernzusteuern.

Ein Doppelklick auf das Symbol oder die Bezeichnung öffnet die entsprechende Maske. Unterprogramme der Masken sind durch das jeweilige Symbol gekennzeichnet. Diese lassen sich generell aus der übergeordneten Maske starten, sind aber für Suchzwecke auch direkt aufrufbar. Ein zugehöriges Favoritenmenü kann sich jeder Benutzer selber einrichten.

Anhand des Versenden eines mails um kurz exemplarisch zu zeigen, wie das funktioniert:

```

Mailstring:='mailto:'+Empfaenger+'&body='+Mailtext;
// Für ShellExecute wird in uses ShellApi verlangt
ShellExecute(handle, 'open', PChar(Mailstring), nil, nil, SW_SHOW);
// Mailfenster als aktuelles Fenster herholen
targetWd:= FindWindowByTitle(Fenstertext);
//Wartezeit bis das Fenster aufgeht.
sleep(2000);
// jetzt noch ALT + S um das Mail zu verschicken.
if targetWd=FindWindowByTitle(Fenstertext) then PostKeyEx32(Ord('S'), [ssAlt], true)
else begin
    sleep(2000);

```

```

targetWd:= FindWindowByTitle(FensterText);
PostKeyEx32(Ord('S'), [ssAlt], true)
end;

```

1. Mail App mit shellexecute() aufrufen
2. Wartezeit einbauen
3. Als sendkey oder postkeyex32 die App fernsteuern
4. Eventuell nochmals versuchen

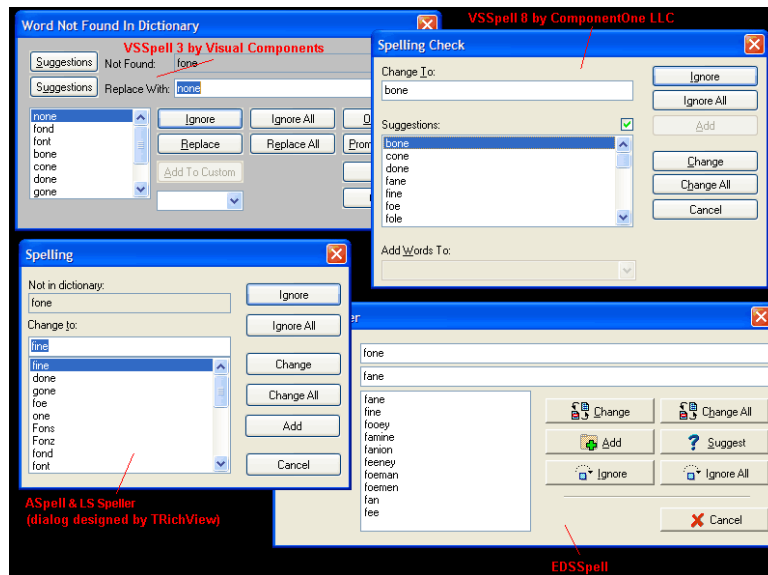


Abb. 2: Spell Checker als Component

## 1.4 OO Component Project

The main goal is to make one or more components for Delphi, with which the developer can easily "talk" with (Star)OpenOffice. The 2nd goal is to make wrapper (for Delphi), which combines the automation components for MS Office and the above.

<http://sourceforge.net/projects/delphi2ooo>

<http://spanish.osstrans.net/software/delphi2ooo.html>

La meta principal es hacer unos o más componentes para Delphi, con el cual del revelador de la poder el " fácilmente; talk" con (estrella) OpenOffice. La 2da meta es hacer la envoltura (para Delphi), que combina los componentes de la automatización para el MS oficina y el antedicho. □

Links:

<http://wiki.services.openoffice.org/wiki/Uno/Delphi>

<http://sourceforge.net/projects/delphi2ooo>

<http://openoffice-delphi-component.qarchive.org/>