

Programmierung von MS-Office mit Visual Basic

Dirk von Hagen

EDV-Beratung Dirk von Hagen
Robert-Mayer-Straße 40
60486 Frankfurt am Main

Inhalt

- Verwendung
- Arten
- Beispiele
- Ausblick
- Historische Anmerkung

Verwendung

- Automatisches Erstellen und Bearbeiten von
 - Dokumenten
 - Tabellen
 - Präsentationen
 - Datenbanken
 - Mails, Kontakten und Terminen
- Steuerung und Anpassung der Anwendungen



Verwendung

- Reagieren auf Ereignisse:
 - Beim Öffnen eines Dokuments wird die Darstellung an den Benutzer angepaßt
 - Beim Schließen eines Dokuments wird der Inhalt geprüft
 - Bei Änderung der Datenbasis wird ein Diagramm aktualisiert

Verwendung

- Datenaustausch zwischen Anwendungen:
 - Bericht in Word entnimmt Daten aus Excel
 - Excel erzeugt Auswertungen von Daten aus einer Access-Datenbank
 - Präsentation entnimmt Texte aus einem Bericht

Visual Basic = Visual Basic ?

- Visual Basic Script (VBScript)
- Visual Basic for Applications (VB)
- Visual Basic (VB)

Visual Basic (VB)

- Eigenständige Programmiersprache (Objektorientiert)
- Eigene Entwicklungsumgebung (Visual Studio)
- Erzeugung von eigenständigen Anwendungen oder Modulen

Visual Basic for Applications (VBA)

- Makro-Sprache von MS-Office
- Programmierschnittstelle der Office-Anwendungen für VB
- Eingebettet in Office -Anwendung:
 - Objekte beziehen sich auf die Anwendung
 - Speicherung in Dokument der Anwendung
 - Entwicklungsumgebung in Anwendung integriert

Visual Basic Script

- Modul des Windows Scripting Hosts
- Aktive Inhalte auf Internet-Seiten
- Makro-Sprache für Administration
- Zugriff auf System-Objekte (Drive, FileSystem, WMI, ADSI, ...)
- Keine Datentypen
- Kein direkter Zugriff auf Office -Objekte

VB / VBA

- VB für eigenständige Anwendungen
- VBA für Tools innerhalb Office -Anwendungen
- Übereinstimmende Syntax
- Übereinstimmendes Objektmodell

Objekte in VB / VBA

Application

Workbook

Document

Presentation

Worksheet

10.04.2003

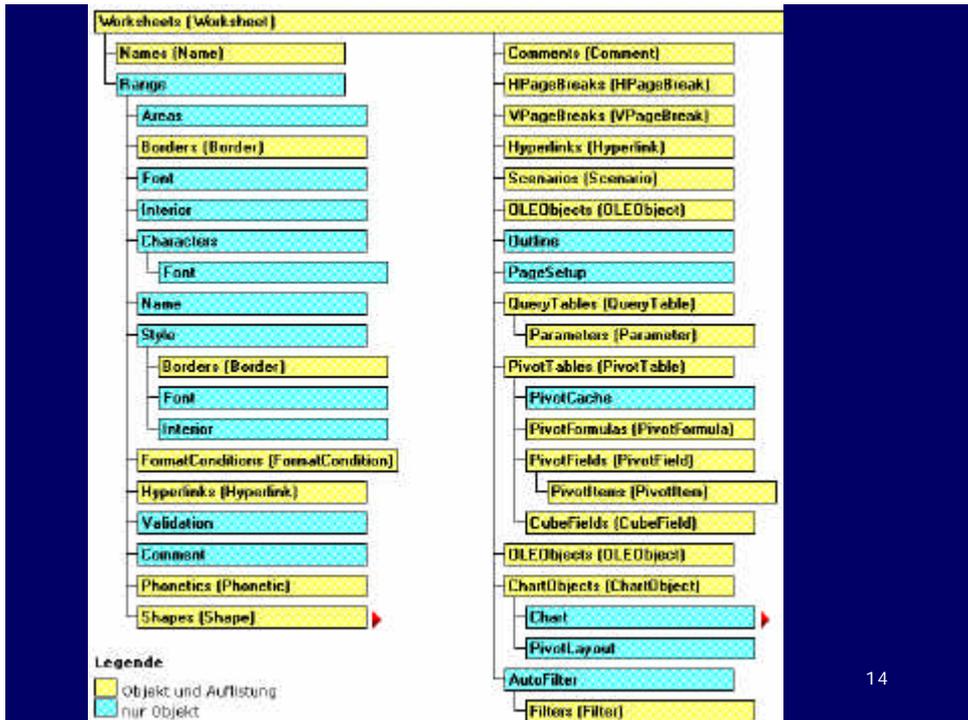
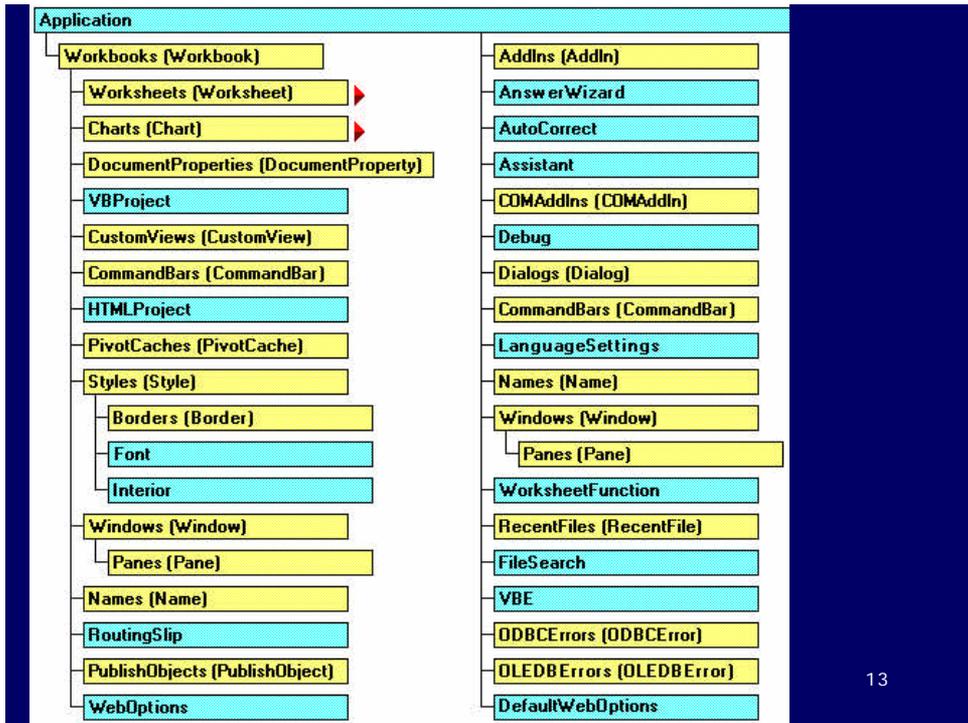
11

Objekte in VB / VBA

- Beispiel Objektbaum von Excel

10.04.2003

12



Beispiel

- Zugriff auf Objekt Application

```
Dim xl As Excel.Application  
Set xl = CreateObject("Excel.Application")  
MsgBox xl.Version
```

- Öffnen einer Excel-Arbeitsmappe

```
xl.Workbooks.Open FileName:=„DECUS.xls“
```

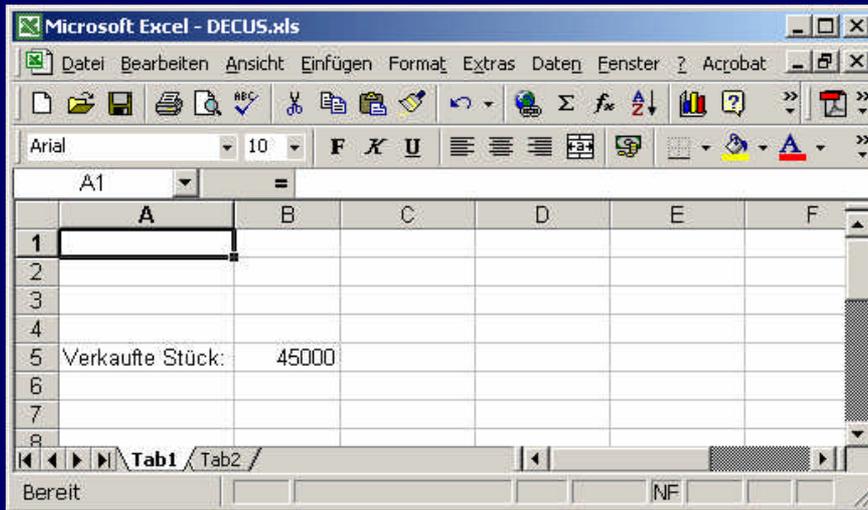
- Schließen einer Excel-Arbeitsmappe

```
xl.Workbooks(„DECUS.xls“).Close  
oder  
xl.ActiveWorkbook.Close
```

Beispiel

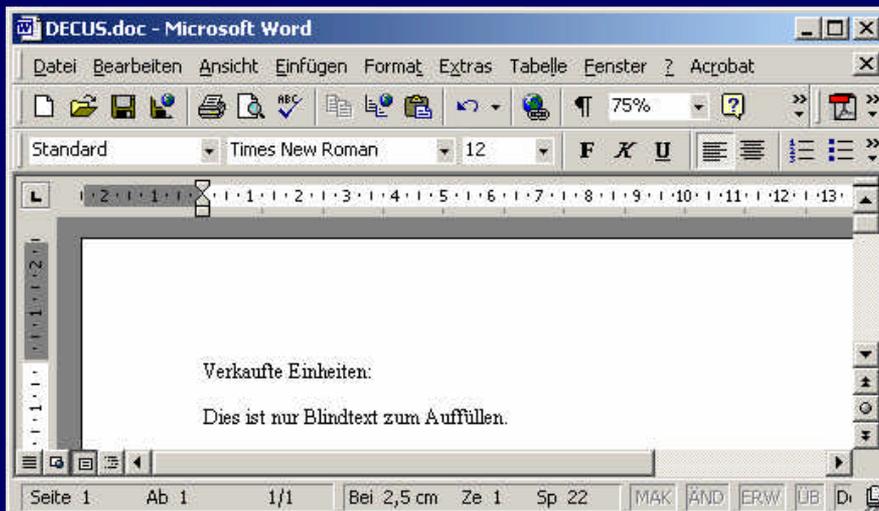
- Daten von Excel zu Word übertragen

```
Set xl = CreateObject(„Excel.Application“)  
Set ww = CreateObject(„Word.Application“)  
  
Set Zelle =  
    xl.Workbooks(„DECUS.xls“).Worksheets(„Tab1“).Range(„B5“)  
  
Set TestPos =  
    ww.Documents(„DECUS.doc“).Paragraphs(1).Range  
  
Text = Zelle.Value  
TestPos.InsertAfter Text
```



10.04.2003

17



10.04.2003

18



10.04.2003

19

Beispiel

- Anpassen der Anwendung

```
Set xl = CreateObject(„Excel.Application“)
```

```
xl.CommandBars(„PDFMaker 5.0“).Enabled = True
```

```
xl.CommandBars(„PDFMaker 5.0“).Visible = True
```

```
xl.CommandBars(„Worksheet Menu Bar“).Enabled = False
```

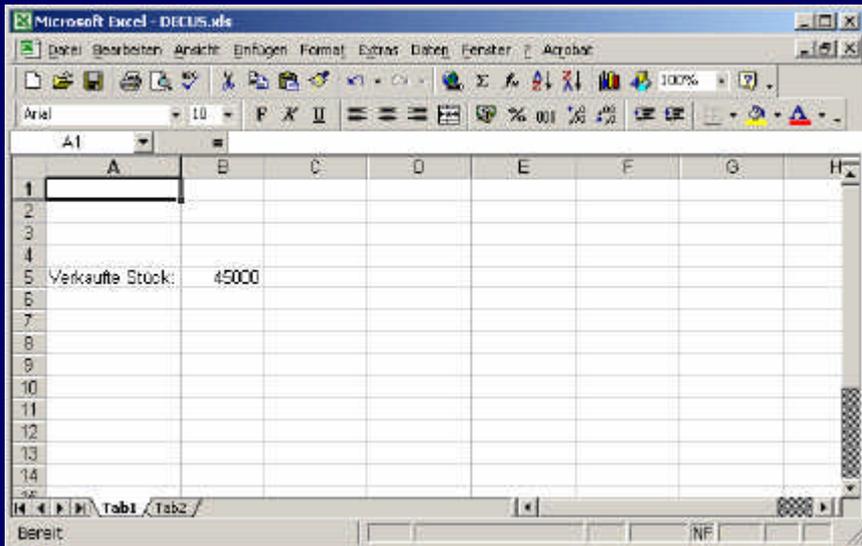
```
xl.DisplayFormulaBar = False
```

```
xl.DisplayStatusBar = False
```

```
xl.Caption = „DECUS“
```

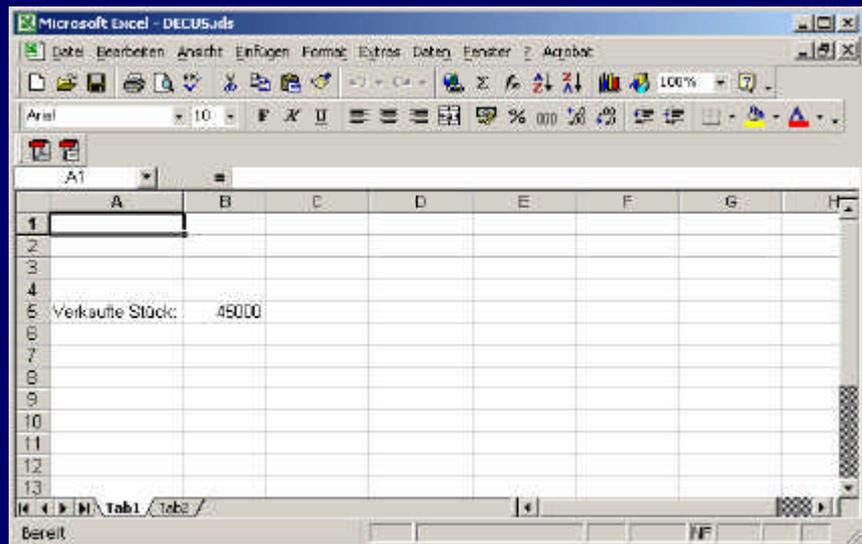
10.04.2003

20



10.04.2003

21



10.04.2003

22

	A	B	C	D	E	F	G	H
1								
2								
3								
4								
5	Verkaufte Stück	45000						
6								
7								
8								
9								
10								
11								
12								
13								
14								
15								
16								
17								
18								
19								
20								
21								

10.04.2003

23

Ausblick MS-Office 2003

- Erweiterte XML-Unterstützung
- Erstellung von „Smart Documents“
- Erweiterungen des Objektmodells

10.04.2003

24

Zusammenfassung

- VB und VBA geeignet für Programmierung von MS-Office
- VB zur Entwicklung eigenständiger Anwendungen
- VBA zur Entwicklung von Tools innerhalb der Office-Anwendungen
- Intuitives Objektmodell
- Das Application-Objekt ist der Ausgangspunkt für den Zugriff auf alle Office-Objekte

„Historische“ Anmerkung

Projekt: Verwaltung von Wirtschaftsplänen

Anforderung:

1. Einfache Erstellung von Wirtschaftsplänen
2. Verwaltung mehrerer Jahre und mehrerer Exemplare
3. Zentrale Kalkulation der Wirtschaftsplanzahlen
4. Automatische Aktualisierung dieser Zahlen in den Dokumenten

„Historische“ Anmerkung

Die Lösung:

1. DCL-Programm:
 - Verwaltung der Wirtschaftspläne der verschiedenen Jahre und Exemplare
 - Steuerung der Textverarbeitung und der Kalkulation
2. DECwrite:
 - Bearbeitung der Dokumente
3. DECcalc:
 - Kalkulation der Wirtschaftsplanaten
4. Fortran-Programm:
 - Konvertierung der Kalkulationszahlen in DECwrite-Variablen

10.04.2003

27

Ende

Vielen Dank für Ihr Interesse.

Haben Sie noch Fragen?

10.04.2003

28