

Makra pro OpenOffice a LibreOffice

PhDr. Mgr. Jeronym Klimeš, Ph.D. 2012

jeronym.klimes@seznam.cz

Kompletně jsem přešel z Windows na Ubuntu, přesněji na Lubuntu, více viz článek [Končíme Bill Gatesi!](#) Musel jsem proto předělat nejnnutnější makra z MS Office do OpenOffice, potažmo do LibreOffice. To bylo asi na celém přechodu z Windows do Linuxu to nejtěžší. Předpokládám proto, že se Vám moje makra mohou hodit. Proto jsem vám exportoval makra a přidal komentář potřebný k pochopení.

Práce se styly a příprava presentace.....	1
Vlastní makra na přepínání stylů.....	2
Importování stylů do aktuálního dokumentu.....	3
Přejmenování starých z stylů z MSWord dokumentů.....	4
Aktualizování stylu podle výběru.....	4
Makra na zvětšování dokumentu.....	5
Makra na formátování aktuálního odstavce.....	5
Pole do Writer na výpočet počtu normovaných stran.....	8
Makra do Calc.....	8

Práce se styly a příprava presentace

Pěstounská péče na přechodnou dobu

přeloženo do češtiny:

Děti jedou na prázdniny ke strýčkovi a tetě na venkov!

Co dělá strýček s tetou:

HONZÍKOVA CESTA



- ukáží dítěti, že mezi mužem a ženou existuje i jiný mód chování než permanentní válka a hádka
- povídají si s dítětem o tom, co zažilo
- řídí styk dítěte s oběma rodiči
- pracují psychologicky/sociálně s oběma rodiči zvlášť
- zprostředkují komunikaci rodičů
- podávají report OSPOD, soudu o chování obou rodičů, aby nezaujal pozorovatele (součást znaleckých posudků).

Živím se přednášením, tedy potažmo výrobou presentací. Zásadně se vyhýbám PowerPointu, ale vždy jsem presentace psal ve Wordu, kde mám stránku na krajinu (landscape) a definované základní styly, viz ukázka vlevo.

Obrázky vkládám vždy jako externí odkaz a vždy jako znak. Toto rozmístění na stránce řeším strukturou řeším tabulkou s neviditelnými kraji (dvě buňky v jedné řádce).

Když je presentace hotová v Wordu/Writeru, tak ji exportuji do PDF. Video pouštím do nového okna.

Ještě z dob, kdy jsem pracoval v AmiPro, tak mám zažito, že skupiny stylů mám pod jednou klávesou: F2 – obyčejné styly, tzn. když máčkám opakovaně F2, tak se přepínají styly odstavců v tomto pořadí:

Výchozí odsazený

Výchozí vlevo

Výchozí vpravo

Výchozí střed

Pod klávesou F3 mám k nim tučné varianty, titulek a mezititulek:

Výchozí tučný odsazený

Výchozí tučný vlevo

Výchozí tučný vpravo

Výchozí tučný střed

Titulek (pod obrázkem)

s následující legendou

Mezititulek, který se nezobrazuje v obsahu

Pod klávesou F4 mám různě číslované a odsazené seznamy a odrážky. Pod klávesou F5 mám nadpisy 1-4, které se zahrnují do obsahu.

Protože takto se píše presentace velice rychle, potřeboval jsem si na toto udělat makra.

Vlastní makra na přepínání stylů

global citac as integer

```
sub nadpisy
styly("nadpisy")
rem Na tato krátká makra se vždy naváže klávesová zkratka (Tools/Customize/Keyboard).
rem V tomto případě klávesa F5
end sub
```

```
sub vychozi
styly("vychozi")
rem Klávesa F2
end sub
```

```
sub tucne
styly("tucne")
rem Klávesa F3
end sub
```

```
sub seznamy
styly("seznamy")
Rem Klávesa F4
end sub
```

```
sub styly(druh as string)
rem toto makro přepíná jednotlivé styly. Používá globální čítač, takže jede vždy od 1, 2,
3 atd.
```

```
rem -----
rem define variables
dim document as object
dim dispatcher as object
dim seznam(7) as string
```

```
if druh="nnadpisy" then
seznam(1)="Vychozi_odsazeny"
seznam(2)="Nadpis 1"
seznam(3)="Nadpis 2"
seznam(4)="Nadpis 3"
seznam(5)="Nadpis 4"
seznam(6)=""
seznam(7)=""
elseif druh="nadpisy" then
seznam(1)="Vychozi_odsazeny"
seznam(2)="Heading 1"
seznam(3)="Heading 2"
seznam(4)="Heading 3"
seznam(5)="Heading 4"
seznam(6)=""
seznam(7)=""
elseif druh="vychozi" then
seznam(1)="Vychozi_odsazeny"
seznam(2)="Vychozi_vlevo"
seznam(3)="Vychozi_stred"
seznam(4)="Vychozi_vpravo"
seznam(5)=""
seznam(6)=""
seznam(7)=""
elseif druh="tucne" then
seznam(1)="MeziTitulek"
seznam(2)="Titulek"
seznam(3)="Tucny_odsazeny"
seznam(4)="Tucny_vlevo"
seznam(5)="Tucny_stred"
seznam(6)="Tucny_vpravo"
```

```
seznam(7)=""
elseif druh="sseznamy" then
seznam(1)="Seznam 1"
seznam(2)="Seznam 2"
seznam(3)="Seznam 3"
seznam(4)="Seznam 4"
seznam(5)="Seznam 5"
seznam(6)="Seznam 6"
seznam(7)=""
elseif druh="seznamy" then
seznam(1)="List 1"
seznam(2)="List 2"
seznam(3)="List 3"
seznam(4)="Numbering 1"
seznam(5)="Numbering 2"
seznam(6)="Numbering 3"
seznam(7)=""
Endif
seznam(0)=""

rem -----
rem get access to the document
document = ThisComponent.CurrentController.Frame
dispatcher = createUnoService("com.sun.star.frame.DispatchHelper")

rem -----
if seznam(citac)="" then citac=1 ' iniciuje čítač, je-li třeba

dim args(1) as new com.sun.star.beans.PropertyValue
args(0).Name = "Template"
args(0).Value = seznam(citac)
args(1).Name = "Family"
args(1).Value = 2

dispatcher.executeDispatch(document, ".uno:StyleApply", "", 0, args())

'resetování čítače, není-li již žádný další styl na přepnutí
if seznam(citac+1)="" then citac = 1 else citac = citac + 1

end sub
```

Importování stylů do aktuálního dokumentu

U dokumentů, které přicházejí od jiných lidí, nemám definované své styly. Tak si je importuji tímto makrem:

```
sub import_stylu ()
dim URL as string
' Zde si musíte zadat cestu k souboru, ve kterém máte uložené své styly:
URL= ConvertToURL("/home/jerom/templates/openoffice/template/styly.odt")
Dim oProps(4) as new com.sun.star.beans.PropertyValue
oProps(0).Name = "LoadTextStyles"
oProps(0).Value = True
oProps(1).Name = "LoadFrameStyles"
oProps(1).Value = true
oProps(2).Name = "LoadPageStyles"
oProps(2).Value = False
oProps(3).Name = "LoadNumberingStyles"
oProps(3).Value = true
oProps(4).Name = "OverwriteStyles"
oProps(4).Value = True
'xray oProps

Dim oDoc As Object
oDoc = thiscomponent.StyleFamilies
'xray oDoc
```

```
oDoc.loadStylesFromURL URL, oProps()
msgbox ("Styly importovany z " & URL)
end sub
```

Přejmenování starých z stylů z MSWord dokumentů

Ve starých dokumentech MSWordu mám jinak nazvané styly, proto bylo třeba makro, které je přejmenuje.

```
sub prejmenuje_stare_styly()
dim Starystyl() as string
dim NovyStyl() as string

StaryStyl()= array("Titulek", "Normální vlevo", "Normální odsazený", "Normální střed",
"Normální vpravo", "Normální tučný odsazený", "Normální tučný střed", "Normální tučný
vpravo", "Normální tučný", "Normální", "Seznam", "Seznam 2", "Seznam 3", "Seznam 4",
"Seznam 5")
NovyStyl() = array("Titulek", "Vychozi_vlevo", "Vychozi_odsazený" , "Vychozi_stred" ,
"Vychozi_vpravo" , "Tucny_odsazený" , "Tucny_stred" , "Tučny_vpravo"
, "Tucny_vlevo" , "Text" , "List 1" , "List 2" , "List 3" , "List 4" ,
"List 5" )

dim oDoc as variant
oDoc=thiscomponent
vFamilies = oDoc.StyleFamilies
vParaStyles = vFamilies.getByname("ParagraphStyles")
s="aqwer"+join(vparastyles.elementnames, "qwer"+"qwer"
rem Ze všech stylů se vytvoří jeden dlouhý řetězec, který je oddělován písmeny „qwer“
rem Takto se dá v řetězci najít již existující styl. Asi se to dá zjistit nějak
elegantněji, ale to už jsem nestudoval.

for i = 0 to ubound(starystyl())
if (InStr (s, "qwer"+starystyl(i)+"qwer")>0) then
if (InStr (s, "qwer"+novystyl(i)+"qwer")>0) then
novystyl(i)=novystyl(i)+"_"
end if
thiscomponent.StyleFamilies.getByname("ParagraphStyles").getbyname(StaryStyl(i)).setname(
NovyStyl(i))
' přejmenuje starý styl na nový
ThisComponent.CurrentController.StatusIndicator.start (" "+ i + ". " + starystyl(i) + "
-> " + novystyl(i),0)
' vypíše info na stavovou řádku, status bar
end if
next i
end sub
```

Aktualizování stylu podle výběru

Ve Wordu stačilo kliknout na styl a zmáčknout enter. Zde je třeba toto makro.

```
sub UpdateCurrentStyleByExample()
rem -----
rem define variables
dim document as object
dim dispatcher as object
dim CurrentPara as string
rem -----
rem get access to the document
document = ThisComponent.CurrentController.Frame
dispatcher = createUnoService("com.sun.star.frame.DispatchHelper")
CurrentPara =ThisComponent.CurrentController.getViewCursor().parastyleName
rem -----
dim args2(1) as new com.sun.star.beans.PropertyValue
args2(0).Name = "Param"
args2(0).Value = CurrentPara
args2(1).Name = "Family"
```

```
args2(1).Value = 2
dispatcher.executeDispatch(document, ".uno:StyleUpdateByExample", "", 0, args2())
end sub
```

Makra na zvětšování dokumentu

Ve Writeru nefungují zkratky ctrl+ ctrl- na zvětšování a zmenšování přiblížení/zoom. Myš s kolečkem nepoužívám, takže opět bylo třeba makro.

```
sub ZoomIn( )
Zoom(3)
rem Klávesová zkratka ctrl+
end sub
```

```
sub ZoomOut( )
Zoom(-3)
rem Klávesová zkratka ctrl-
end sub
```

```
function Zoom(SStep as Integer)
rem -----
rem define variables
dim Magnification as Integer
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")

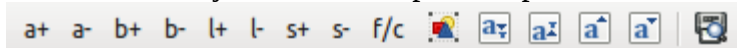
rem -----
Magnification= thiscomponent.currentcontroller.viewsettings.zoomvalue()
Magnification = Magnification + SStep
if (Magnification < 20) then Magnification =20
'restore counting from 100
if ((Magnification > (100-abs(sstep))) and (Magnification < (100+abs(sstep)))) then
Magnification =100
if (Magnification >600) then Magnification =600
'statusbar
ThisComponent.CurrentController.StatusIndicator.setText("Magnification: " +
Magnification)
'print typename(magnification)
dim args1(2) as new com.sun.star.beans.PropertyValue
args1(0).Name = "Zoom.Value"
args1(0).Value = Magnification
args1(1).Name = "Zoom.ValueSet"
args1(1).Value = 28703
args1(2).Name = "Zoom.Type"
args1(2).Value = 0

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

end function
```

Makra na formátování aktuálního odstavce

Používání universálních stylů nestačí, občas je třeba trochu proložit písmo, zvětšit mezeru za odstavcem ap. K tomu jsem si udělal toolbar:



a+/a- mění mezeru za odstavcem

b+/b- mění mezeru před odstavcem

l+/l- mění řádkování

s+/s- mění proložení znaků

f/c – mění jemnost hrubost kroku – tedy skáče po malých či velkých krocích

```

Global fine as integer
global faktor(3) as integer

'option explicit
sub statusbar(txt as string)
ThisComponent.CurrentController.StatusIndicator.start(txt,50)
rem pomocná rutina na výpis na stavový řádek
end sub

sub switch_fine
rem tlačítko f/c
fine =fine +1
if fine > 2 then fine=0

statusbar("Fine/coarse factor: " + (fine+1))
' ThisComponent.CurrentController.StatusIndicator.start("Fine/coarse factor: " +
  (fine*-9+1),50)
end sub

sub linespacingup
rem tlačítko l+
linespacing(+1)
end sub

sub linespacingdown
rem tlačítko l-
linespacing(-1)
end sub

sub linespacing(updown)
'print isarray(faktor)
faktor=array(1, 5, 50)
'print faktor(fine)
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 get values-----
oCurSelection = ThisComponent.getCurrentSelection()
if oCurSelection.supportsService("com.sun.star.text.TextRanges") then
  nCount = oCurSelection.Count
  if nCount = 1 then
    ' get the text object of the selection
    oTextRange = oCurSelection.getByIndex(0)
    ' xray otextrange
    'print oTextRange.paratopmargin
    'print before
    'print oTextRange.parabottommargin
    'print after

    percentage= otextrange.paralinespacing.height + updown*faktor(fine)

    statusbar("paralinespacing: " + percentage)

rem -----
dim args1(1) as new com.sun.star.beans.PropertyValue
args1(0).Name = "LineSpacing.Mode"
args1(0).Value = 0
args1(1).Name = "LineSpacing.Height"
args1(1).Value = percentage

```

```

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

end if
end if
end sub

sub beforeup()
rem tlačítka b+
predpoparagrafem(1, 0)
end sub

sub beforedown()
rem tlačítka b-
predpoparagrafem(-1,0)
end sub

sub afterup()
rem tlačítka a+
predpoparagrafem(0, 1)
end sub

sub afterdown()
rem tlačítka a-
predpoparagrafem(0,-1)
end sub

sub predpoparagrafem(before, after)
faktor=array(1, 5, 50)
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 get values -----
oCurSelection = ThisComponent.getCurrentSelection()
if oCurSelection.supportsService("com.sun.star.text.TextRanges") then
  nCount = oCurSelection.Count
  if nCount = 1 then
    ' get the text object of the selection
    oTextRange = oCurSelection.getByIndex(0)
    topmargin = oTextRange.paratopmargin + before*faktor(fine)
    if topmargin<0 then topmargin=0
    bottommargin= oTextRange.parabottommargin + after*faktor(fine)
    if bottommargin<0 then bottommargin=0
    'status bar message:
    statusBar("top margin: " + topmargin +"; bottom margin: " + bottommargin)
  rem -----
  dim args1(3) as new com.sun.star.beans.PropertyValue
  args1(0).Name = "TopBottomMargin.TopMargin"
  args1(0).Value = topmargin
  args1(1).Name = "TopBottomMargin.BottomMargin"
  args1(1).Value = bottommargin
  args1(2).Name = "TopBottomMargin.TopRelMargin"
  args1(2).Value = 100
  args1(3).Name = "TopBottomMargin.BottomRelMargin"
  args1(3).Value = 100

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

end if
end if
end sub

```

```

sub scaleup()
rem tlačítko s+
Scale(1)
end sub

Sub scaledown()
rem tlačítko s-
Scale(-1)
end sub

sub Scale(updown as integer)
faktor=array(1, 3, 10)
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 -----
'if ScalingFactor =0 then scalingFactor =100
'ScalingFactor = ScalingFactor + updown

oCurSelection = ThisComponent.getCurrentSelection()
if oCurSelection.supportsService("com.sun.star.text.TextRanges") then
nCount = oCurSelection.Count
if nCount = 1 then
' get the text object of the selection
oTextRange = oCurSelection.getByIndex(0)
ScalingFactor= oTextRange.Charscalewidth + updown*faktor(fine)
statusbar("CharacterWidthScalingFactor: " + ScalingFactor,50)
rem -----
dim arg1(0) as new com.sun.star.beans.PropertyValue
arg1(0).Name = "CharacterWidthScalingFactor"
arg1(0).Value = ScalingFactor
dispatcher.executeDispatch(document, ".uno:CharacterWidthScalingFactor", "", 0, arg1())
END IF
END IF

end sub

```

Pole do Writer na výpočet počtu normovaných stran

Do záhlaví dokumentu si vkládám toto pole, které mi někdo poradil:

Mělo by stačit vložit po stisknutí F2:

=char/1800 round 1

kde 1 je počet desetinných míst, na který se bude zaokrouhlovat.

Makra do Calc

Často potřebuji přičítat/odčítat jedničku k nějaké hodnotě v buňce. Tak jsem si napsal makra a přiřadil ke klávesám.

```

sub jednaplus
rem ctrl-shift-J
jednaplusminus(1)
end sub

```

```

sub jednaminus
rem ctrl-shift-K
jednaplusminus(-1)
end sub

```



```
sub jednaplusminus(prirustek)
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")

'get current cell value
oCell = ThisComponent.CurrentController.Selection

rem -----
dim args1(0) as new com.sun.star.beans.PropertyValue
args1(0).Name = "StringName"
args1(0).Value = (oCell.DataArray(0)(0) + prirustek)&"
dispatcher.executeDispatch(document, ".uno:EnterString", "", 0, args1())

end sub
```