"Automating OpenOffice with ooRexx: ooRexx Nutshell Examples for Write and Calc and..."

2005 International Rexx Symposium
Los Angeles, California, U.S.A. (April 2005)

Rony G. Flatscher (Rony.Flatscher@wu-wien.ac.at)
Wirtschaftsuniversität Wien, Austria (http://www.wu-wien.ac.at)
Agenda

• Overview
• Easying the interfacing with OOo
  – OOo.cls
    • Developed while at the symposium
    • Needs to be extended
      – Anyone is invited to do so and share the results!
• Roundup and Outlook
Sources of figures, examples and hints


• Mr. Augustin's paper "Erweiterung der Skriptfähigkeit von OpenOffice.org durch BSF und JSR-223" at the "WU Wien", cf. http://www.matt.at/oo_examples

/* initialize connection to server, get its Desktop-service and XComponentLoader interface */
xComponentContext = .bsf~new("com.sun.star.comp.helper.Bootstrap") ~createInitialComponentContext(.nil)
xUrlResolver = xComponentContext~getServiceManager() ~createInstanceWithContext("com.sun.star.bridge.UnoUrlResolver", xComponentContext)

unoResolverName = .bsf4rexx~Class.class~forName("com.sun.star.bridge.XUnoUrlResolver")
unoRuntime = .bsf~new("com.sun.star.uno.UnoRuntime")
urlResolver = unoRuntime~queryInterface(unoResolverName, xUrlResolver)

unoUrl = "uno:socket,host=localhost,port=8100;urp;StarOffice.NamingService"
rInitialObject = urlResolver~resolve(unoUrl)
namingServiceName = .bsf4rexx~Class.class~forName("com.sun.star.uno.XNamingService")
rName = unoRuntime~queryInterface(namingServiceName, rInitialObject)
rXsmgr = rName~getRegisteredObject("StarOffice.ServiceManager")
msfName = .bsf4rexx~Class.class~forName("com.sun.star.lang.XMultiServiceFactory")
xMsf = unoRuntime~queryInterface(msfName, rXsmgr)

-- Retrieve the Desktop object, we need its XComponentLoader interface
-- to load a new document
aDesktop = xMsf~createInstance("com.sun.star.frame.Desktop")
xDesktop = .bsf4rexx~Class.class~forName("com.sun.star.frame.XDesktop")
oDesktop = unoRuntime~queryInterface(xDesktop, aDesktop)
xComponentLoaderName = .bsf4rexx~Class.class~forName("com.sun.star.frame.XComponentLoader")
xComponentLoader = unoRuntime~queryInterface(xComponentLoaderName, oDesktop)

/* Open a blank text document */
/* No properties needed */
propertyValueName = .bsf4rexx~Class.class~forName("com.sun.star.beans.PropertyValue")
loadProps = .bsf~createArray(propertyValueName, 0) /* 0=no elements, i.e. empty Java array */
/* load an empty text document */
xWriterComponent = xComponentLoader~loadComponentFromURL("private:factory/swriter", "_blank", 0, loadProps)
::requires "BSF.cls"
• Initializing OOo a recurrent issue
  – Load off the needed statements
• Support an OOo-proxy
  – Makes it easy to get XInterfaces from the objects
  – Works closely with BSF
    • Wraps up BSF proxies
• Eases coding of OOo considerably
/* initialize connection to server, get its Desktop-service and XComponentLoader interface */
xMsf=ooo.connect() -- connect to server and retrieve remote multi server factory

-- Retrieve the Desktop object, we need its XComponentLoader interface
-- to load a new document
oDesktop = xMsf~createInstance("com.sun.star.frame.Desktop")
xDesktop = oDesktop~XDesktop -- get desktop interface
xComponentLoader = xDesktop~XComponentLoader -- get componentLoader interface

/* load an empty text document */
xWriterComponent = xComponentLoader~loadComponentFromURL("private:factory/swriter", "_blank", 0, .OOo~noProps)
::requires "OOo.cls" -- get Oo support
Roundup and Outlook

• OOo
  – Needs BSF4Rexx
  – Full control over Open Office

• OOo.cls
  – Eases programming considerably
    • Making it easy to request interface objects
  – Needs enhancements
    • Please share yours!

• You can directly apply all OOo information
  – StarBasic documentation, books
  – UNO documentation, books for C++, Java

• With OOo v2.0
  – Will be possible to get ooRexx be used from within OOo, if someone adds the necessary bridging code (in Java)