



# OpenOffice.org Extensions development in Java with NetBeans in practise

Jürgen Schmidt

OpenOffice.org

Sun Microsystems, Inc.





# OpenOffice.org Extensions development in Java with NetBeans in practise

- Motivation
- Introduction
- Demo I: Smart Tag Extension
- Demo II: Addon with Option page
- More Information



# Motivation

## for OpenOffice.org Extensions

- Growing popularity of ODF worldwide
- Growing popularity of OpenOffice.org
- Multi platform support
  - > Solaris/Open Solaris, Linux, Windows, Mac OS, ...
- Demand for customization, extending or integration of the OpenOffice.org
- Community building
  - > lower entry barrier
  - > fast feeling of success



# Introduction

## OpenOffice.org API plugin for NetBeans

- New specialized project types
  - > Wizards to speed up development
  - > Support for UNOIDL and configuration files
  - > Context sensitive API help
  - > Code completion
  - > Debug support
  - > Pre-configured OpenOffice.org library
  - > Online update



# Introduction

## OpenOffice.org API plugin for NetBeans

- UNO client application project
  - > Primary a normal Java 2 Platform, Standard Edition (J2SE™ platform) project
  - > Special packaging
    - Bundle UNO bootstrap glue code from the SDK
      - Search default office, Customized classloader
    - Special manifest entries
  - > Provide remote Office context
  - > Office start on demand
  - > Named pipe connection
  - > Typical use case – remote control of OpenOffice.org



# Introduction

## OpenOffice.org API plugin for NetBeans

- Common for all UNO component project types
  - > Specialized J2SE class library project
  - > Generated code skeletons
  - > Completely functional, buildable out of the box
  - > Special packaging support
  - > Office extension packages (.oxt)
  - > Debug support
  - > Final deployment in target office
  - > Project properties for
    - Extension Identifier, Dependencies and Licenses



# Introduction

## OpenOffice.org API plugin for NetBeans

- Calc Add-in project
  - > High-level definition of new built-in Calc function
    - Abstraction from underlying technology
  - > Localization support
    - Function, parameter names and descriptions
- Completely functional skeleton
  - > Generated code for Calc add-in specific interfaces
  - > Add-in functions default implemented



# Introduction

## OpenOffice.org API plugin for NetBeans

- Add-on project wizard
  - > User interface integration
  - > High-level definition of
    - New commands
    - Top level menu with menu entries and/or sub-menus
    - Toolbar with simple button controls
  - > Localization support
  - > Menu and toolbar entries
  - > Default implementation for add-on specific interfaces
    - SPI: `com.sun.star.frame.ProtocolHandler`





# Introduction

## OpenOffice.org API plugin for NetBeans

- UNO Component project wizard
  - > Integrated UNOIDL wizard
    - Definition of new type definitions
    - Type browser for existing Office types
      - Reuse in new types
      - Implementation of existing services and/or interfaces
  - > Completely functional code skeletons
    - Generated code for component specific interfaces
    - Default implementation for interface methods
- New File Type – Java UNO Object
  - > similar to the Component project wizard



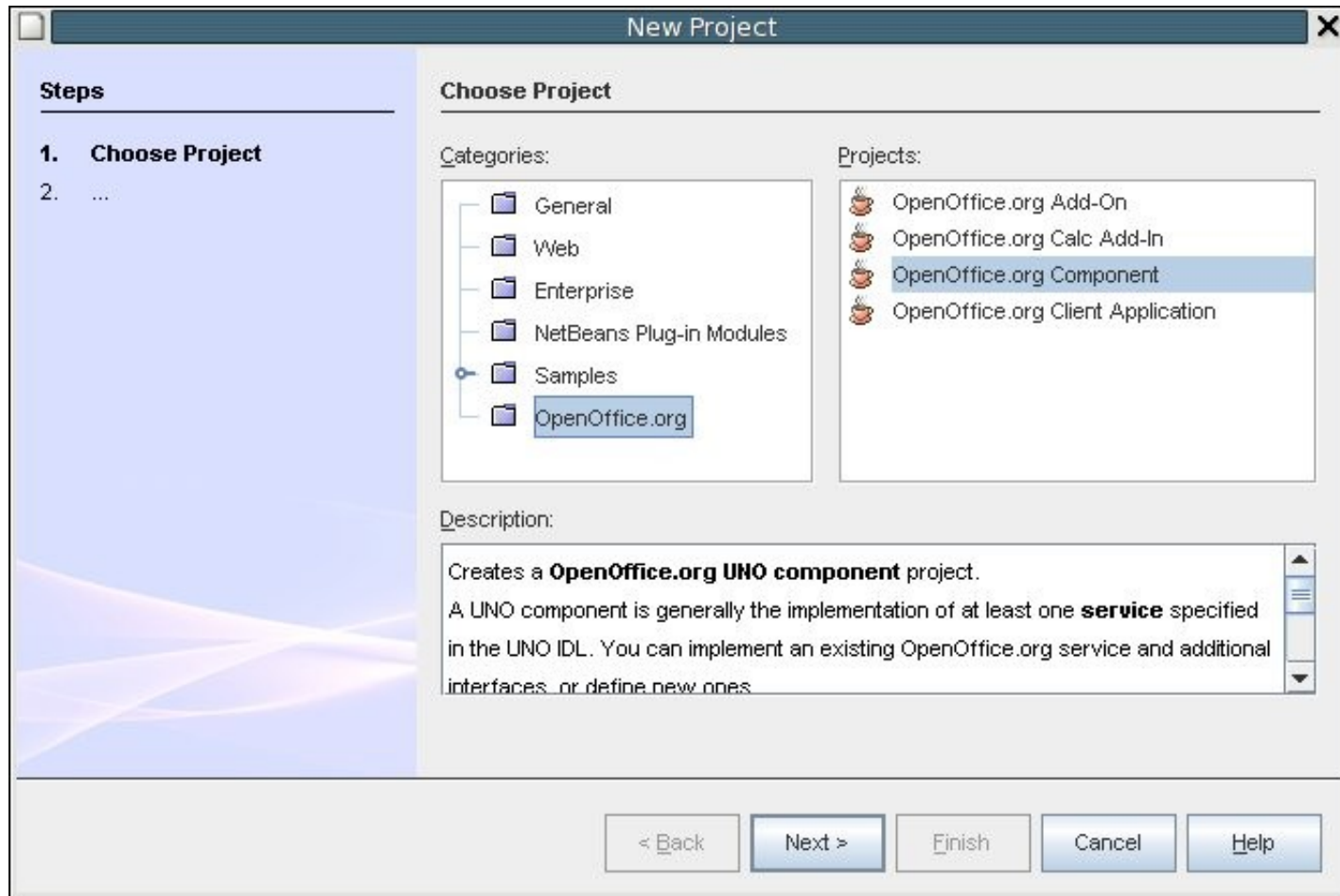
# Demo I: OOIssue Smart Tag

- Java UNO component Extension
- Implements two objects
  - > OOIssueSmartTag recognizer
    - `com.sun.star.smarttag.SmartTagRecognizer`
    - detect OpenOffice.org Issues in the form of `i94678`
    - no properties dialog
  - > OOIssueSmartTagAction
    - `com.sun.star.smarttag.SmartTagAction`
    - two actions are be implemented
      - View in Browser
      - Replace by URL



# Demo I: OOIssue Smart Tag

- Select File → New Project → OpenOffice.org Component





# Demo I: OOIssue Smart Tag

- Specify your project properties

**Steps**

1. Choose Project
- 2. Name and Location**
3. Define Service

**Name and Location**

Project Name:

Class Name:

Package:

Project Location:

Project Folder:

Created Files: src/org/openoffice/demo/OOIssueSmartTagImpl.java  
src/uno-extension-manifest.xml  
(...)

< Back    Next >    Finish    Cancel    Help



# Demo I: OOIssue Smart Tag

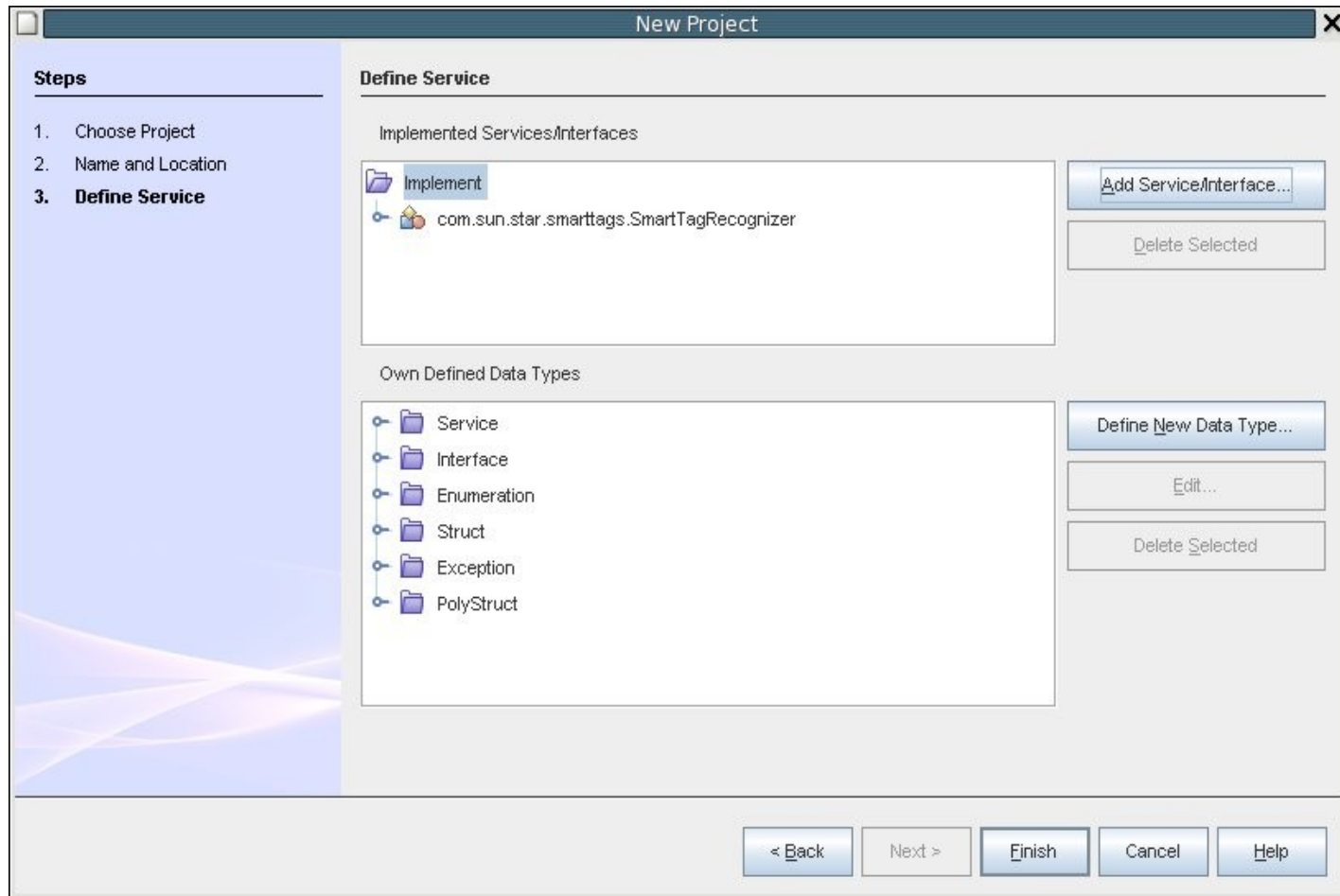
- Select the `com.sun.star.smarttags.SmartTagRecognizer`

The screenshot shows the 'New Project' dialog box with the 'Define Service' tab selected. The 'Implemented Services/Interfaces' section shows a tree view with 'Implement' selected. A 'Select a new Data type' dialog is open, showing a tree view with 'SmartTagRecognizer' selected. The 'Display Type' is set to 'All'. The 'SmartTagRecognizer' is highlighted in blue. The 'SmartTagRecognizer' is selected. The 'Display Type' is set to 'All'. At the bottom of the main dialog, there is a red error message: 'There has to be at least one implemented service with at least one valid interface.'



# Demo I: OOIssue Smart Tag

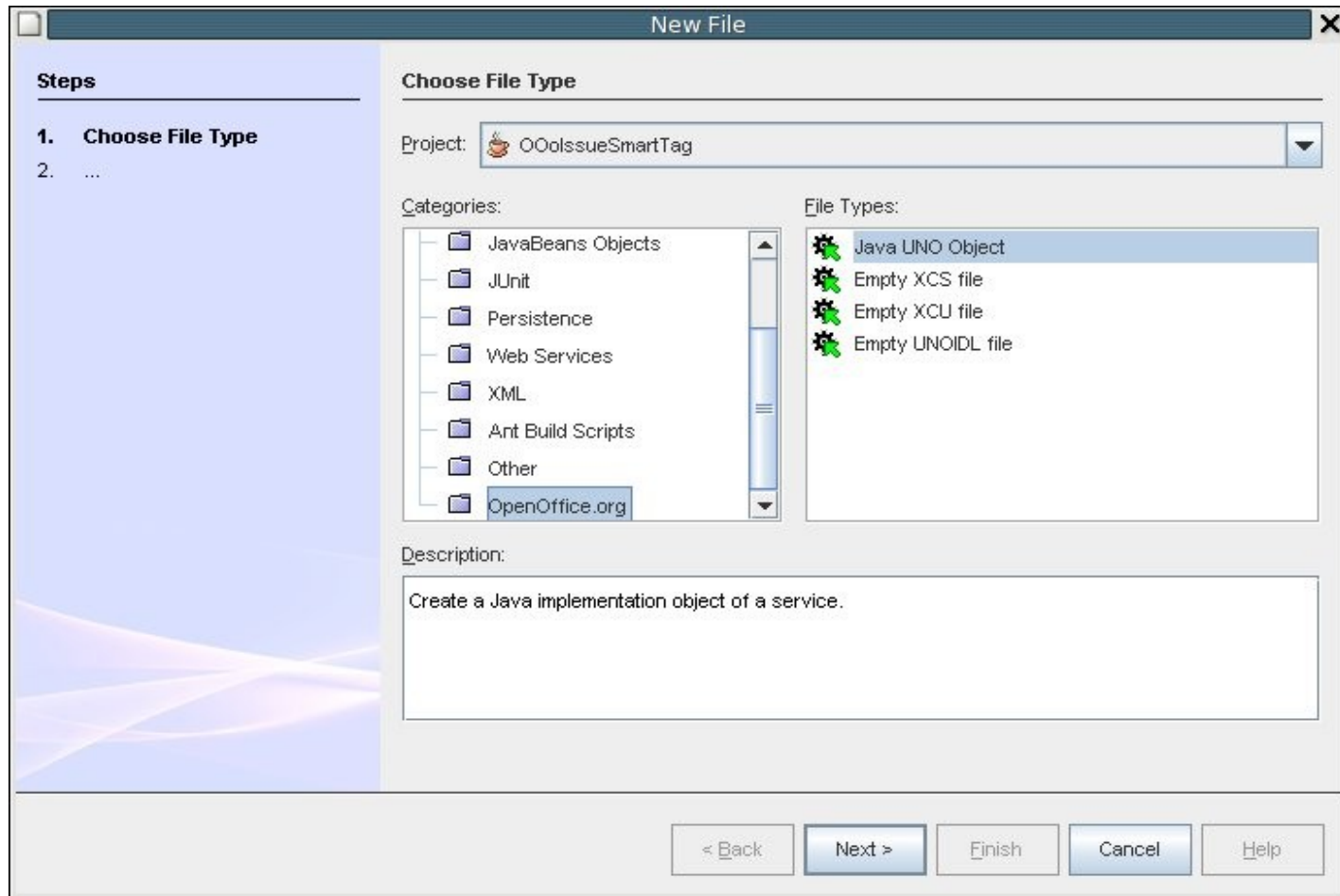
- Finish the wizard to create the new project





# Demo I: OOIssue Smart Tag

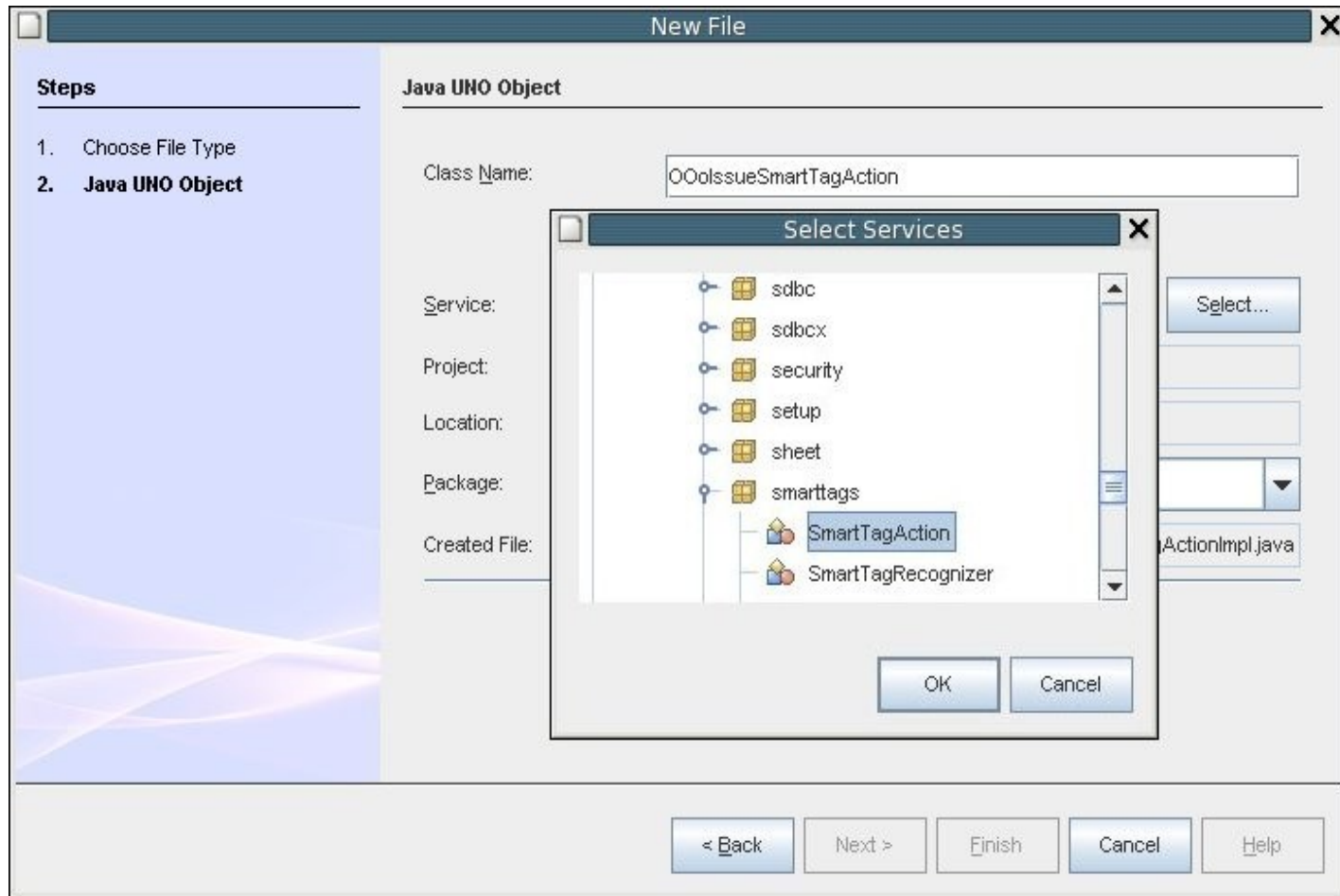
- Create a second UNO implementation object





# Demo I: OOIssue Smart Tag

- Select `com.sun.star.smarttags.SmartTagAction` and finish







# Demo I: OOoIssue Smart Tag

- Extend the generated SmartTagRecognizer object

```

...
private static final String m_smartTagType =
    "org.openoffice#oooissue";

public static String getOOoIssueSmartTagName () {
    return m_smartTagType;
}

...
// com.sun.star.smarttags.XSmartTagRecognizer:
public int getSmartTagCount() {
    return 1;
}

// let these function default implemented
public String getName(Locale aLocale)
public String getDescription(Locale aLocale)
...

```



# Demo I: OOIssue Smart Tag

- Extend the generated SmartTagRecognizer object

```

...
public String getSmartTagName(int nSmartTagIndex)
    throws com.sun.star.lang.IndexOutOfBoundsException {
    return m_smartTagType;
}

// let these function default implemented
public String getSmartTagDownloadURL(int nSmartTagIndex)
public boolean hasPropertyPage(...)
public void displayPropertyPage()
    
```



# Demo I: OOIssue Smart Tag

- Extend the generated SmartTagRecognizer object

```

...
public void recognize(...) {
    final int nEndPos = nStart + nLength;
    Boundary aWordBounds = xTokenizer.getWordBoundary(
        aText, nStart, aLocale, WordType.DICTIONARY_WORD, true );

    while ( aWordBounds.startPos < aWordBounds.endPos &&
        aWordBounds.endPos <= nEndPos ) {
        final String aWord = aText.substring(aWordBounds.startPos,
            aWordBounds.endPos);
        if ( aWord.matches("i\\d{5,6}") ) {
            XStringKeyMap xProps = null;
            xTextMarkup.commitTextMarkup( TextMarkupType.SMARTTAG,
                m_smartTagType, aWordBounds.startPos,
                aWordBounds.endPos - aWordBounds.startPos, xProps );
        }
        // continue with next word
        aWordBounds = xTokenizer.nextWord( aText, aWordBounds.startPos,
            aLocale, WordType.DICTIONARY_WORD );
    }
}

```



# Demo I: OOIssue Smart Tag

- Extend the generated SmartTagAction object

```

...
private final int m_idBase = this.hashCode();
private static final String[] m_actionNames = {
    "oooIssueViewAction", "oooIssueReplaceByUrlAction" };
private static final String[] m_actionCaptions = {
    "View Issue in Browser", "Replace by Url" };

...
// com.sun.star.smarttags.XSmartTagAction:
public int getSmartTagCount() {
    return 1;
}

// let these functions default implemented
public String getName(com.sun.star.lang.Locale aLocale)
public String getDescription(com.sun.star.lang.Locale aLocale)

public String getSmartTagName(int nSmartTagIndex)
    throws com.sun.star.lang.IndexOutOfBoundsException {
    return OOoIssueSmartTag.getOOoIssueSmartTagName();
}
...

```



# Demo I: OOIssue Smart Tag

- Extend the generated SmartTagAction object

```

...
public String getSmartTagCaption(int nSmartTagIndex, Locale aLocale)
    throws com.sun.star.lang.IndexOutOfBoundsException {
    return "OpenOffice.org Issue";
}
public int getActionCount(String aSmartTagName, XController xController) {
    return 2;
}
public int getActionID(String aSmartTagName, int nActionIndex,
    XController xController) throws com.sun.star.lang.IllegalArgumentException {
    return m_idBase + nActionIndex;
}

public String getActionCaptionFromID(int nActionID, ...)
    throws com.sun.star.lang.IllegalArgumentException {
    return m_actionCaptions[nActionID - m_idBase];
}

public String getActionNameFromID(int nActionID, XController xController)
    throws com.sun.star.lang.IllegalArgumentException {
    return m_actionNames[nActionID - m_idBase];
}

```



# Demo I: OOIssue Smart Tag

- Extend the generated SmartTagAction object

```

...
public void invokeAction(int nActionID, String aApplicationName,
    com.sun.star.frame.XController xController,
    com.sun.star.text.XTextRange xTarget,
    com.sun.star.container.XStringKeyMap xProperties, String aText,
    String aXML, com.sun.star.lang.Locale aLocale)
    throws com.sun.star.lang.IllegalArgumentException {
    final int index = nActionID - m_idBase;

    if (index == 0) {
        viewAction(aApplicationName, xController, xTarget, xProperties,
            aText, aXML, aLocale);
    } else if (index == 1) {
        replaceAction(aApplicationName, xController, xTarget, xProperties,
            aText, aXML, aLocale);
    }
}

// let these function default implmented
public boolean isCaptionDynamic(...)
public boolean isShowSmartTagIndicator(...)

```



# Demo I: OOIssue Smart Tag

- Extend the generated SmartTagAction object

```

...
private void viewAction(String aApplicationName,
    com.sun.star.frame.XController xController,
    com.sun.star.text.XTextRange xTarget,
    com.sun.star.container.XStringKeyMap xProperties, String aText,
    String aXML, com.sun.star.lang.Locale aLocale)
    throws com.sun.star.lang.IllegalArgumentException {

    try {
        final XMultiComponentFactory xFact = m_xContext.getServiceManager();
        final Object xObject = xFact.createInstanceWithContext(
            "com.sun.star.system.SystemShellExecute", m_xContext);
        final XSystemShellExecute xSystemShellExecute = (XSystemShellExecute)
            UnoRuntime.queryInterface(XSystemShellExecute.class, xObject);
        String aURLString = "http://www.openoffice.org/issues/show_bug.cgi?id="
            + xTarget.getString().substring(1);
        xSystemShellExecute.execute( aURLString, "",
            com.sun.star.system.SystemShellExecuteFlags.DEFAULTS );
    } catch (Exception ex) {
        ex.printStackTrace();
    }
}

```



# Demo I: OOIssue Smart Tag

- Extend the generated SmartTagAction object

```

...
private void replaceAction(String aApplicationName,
    com.sun.star.frame.XController xController,
    com.sun.star.text.XTextRange xTarget,
    com.sun.star.container.XStringKeyMap xProperties, String aText,
    String aXML, com.sun.star.lang.Locale aLocale)
    throws com.sun.star.lang.IllegalArgumentException {

    String aURLString = "http://www.openoffice.org/issues/show_bug.cgi?id=" +
        xTarget.getString().substring(1);

    XPropertySet xProp = (XPropertySet)UnoRuntime.queryInterface(
        XPropertySet.class, xTarget);

    if (xProp != null) {
        try {
            xProp.setPropertyValue("HyperLinkURL", aURLString);
        } catch (Exception ex) {
            ex.printStackTrace();
        }
    }
}

```





# Demo

**Step By Step example**

**OOI Issue Smart Tag  
Extension**



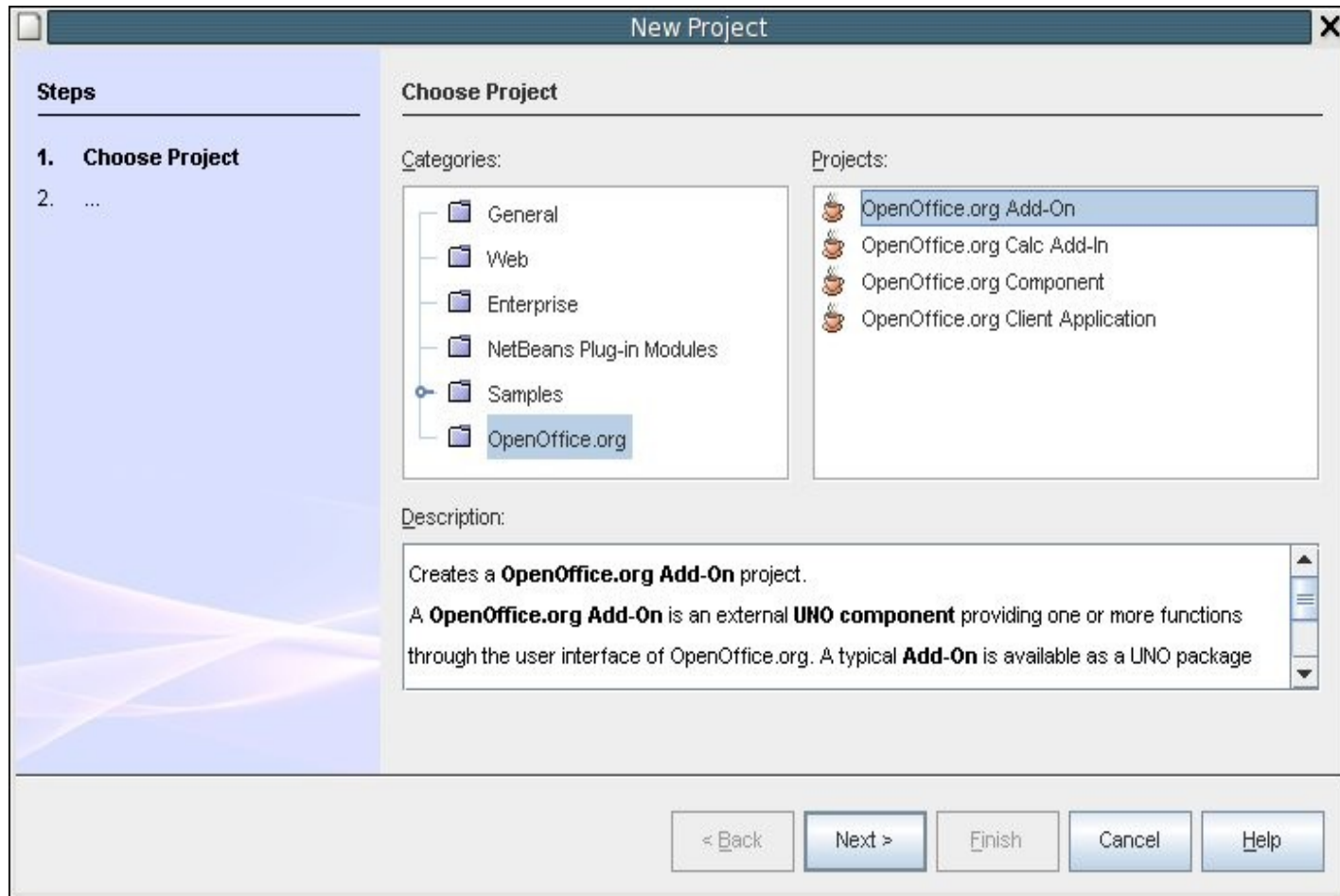
# Demo I: Addon with Option page

- Simple Add-On example
- Options page
  - > Two configuration options
    - Message
    - ColoredMessage
- New menu and toolbar
  - > Both with a “Insert Message” command
    - Insert the configured message at the end of the document
    - Text color depends on the “Colored Message” flag
  - > Visible in the Writer context only



# Demo I: Addon with Option page

- Select File → New Project → OpenOffice.org Add-On





# Demo I: Addon with Option page

- Specify your project properties

New Project
✕

**Steps**

1. Choose Project
- 2. Define Add-On Name and Location**
3. Define User Commands
4. Create UI structure

**Define Add-On Name and Location**

Project Name:

Main Class Name:

Java Package:

Project Location:

Project Folder:

Create Menu

Create Toolbar

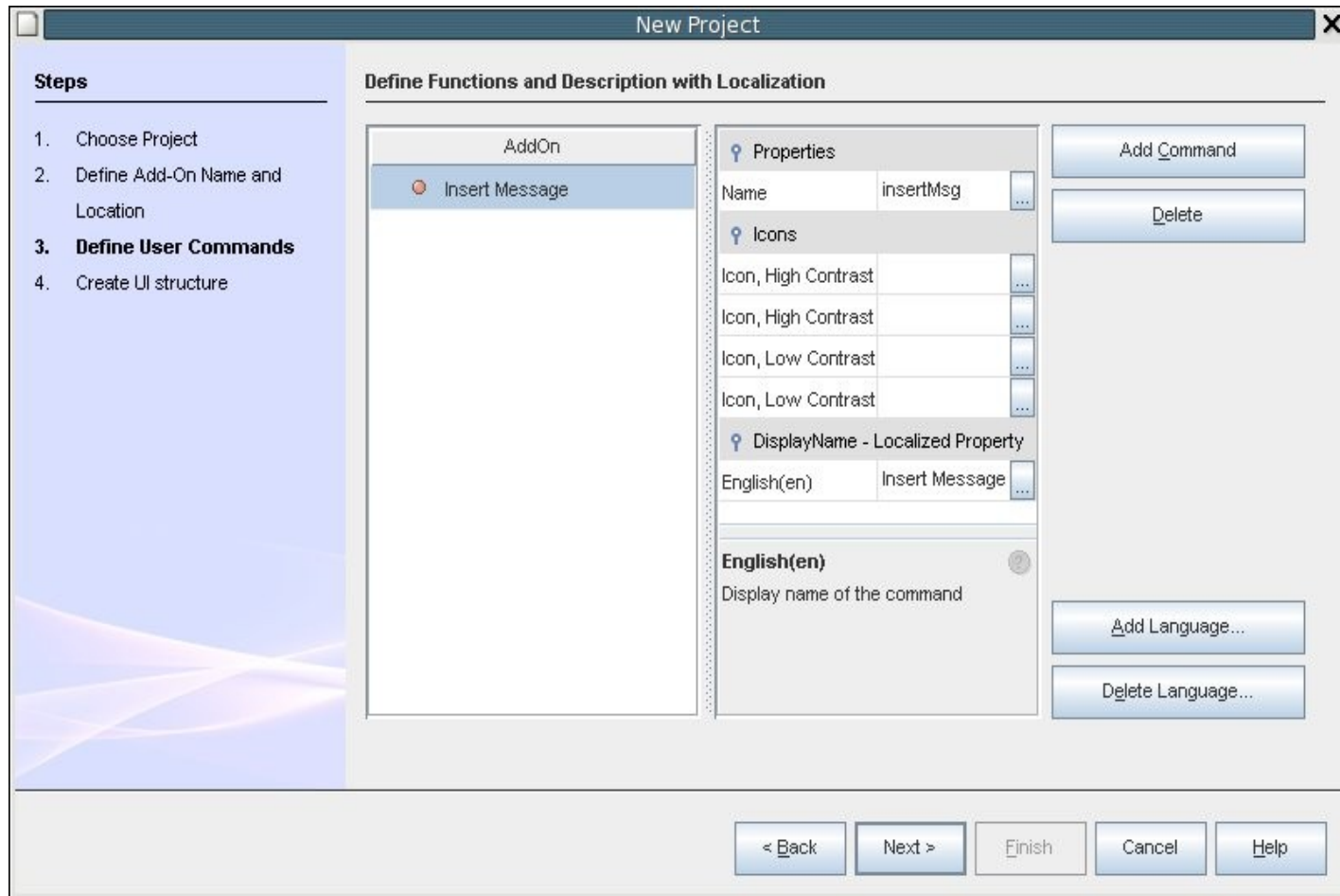
Created Files:

```
src/org/openoffice/demo/WorkshopOptionsDemo.java
src/uno-extension-manifest.xml
src/ProtocolHandler.xcu
src/Addons.xcu
```



# Demo I: Addon with Option page

- Define a new command and localized display names





# Demo I: Addon with Option page

- Define a new menu and arrange your commands

**New Project** [X]

**Steps**

1. Choose Project
2. Define Add-On Name and Location
3. Define User Commands
4. **Create UI structure**

**Define Functions and Description with Localization**

Preview: All

**AddOn**

- Workshop Demo
  - Insert Message

**Properties**

Name: insertMsg

**Context**

Bibliography	<input type="checkbox"/>
Chart	<input type="checkbox"/>
Draw	<input type="checkbox"/>
Formula	<input type="checkbox"/>
Presentation	<input type="checkbox"/>
Spreadsheet	<input type="checkbox"/>
Writer	<input checked="" type="checkbox"/>

**Buttons:** Add Menu, Add Separator, Add Command, Delete

**Navigation:** < Back, Next >, Finish, Cancel, Help



# Demo I: Addon with Option page

- Define a new toolbar and arrange your commands

**Steps**

1. Choose Project
2. Define Add-On Name and Location
3. Define User Commands
4. **Create UI structure**

**Define Functions and Description with Localization**

Preview: All

Insert Message

Add Separator

Add Command

Delete

**Properties**

Context	
Bibliography	<input type="checkbox"/>
Chart	<input type="checkbox"/>
Draw	<input type="checkbox"/>
Formula	<input type="checkbox"/>
Presentation	<input type="checkbox"/>
Spreadsheet	<input type="checkbox"/>
<b>Writer</b>	<input checked="" type="checkbox"/>

**Writer**

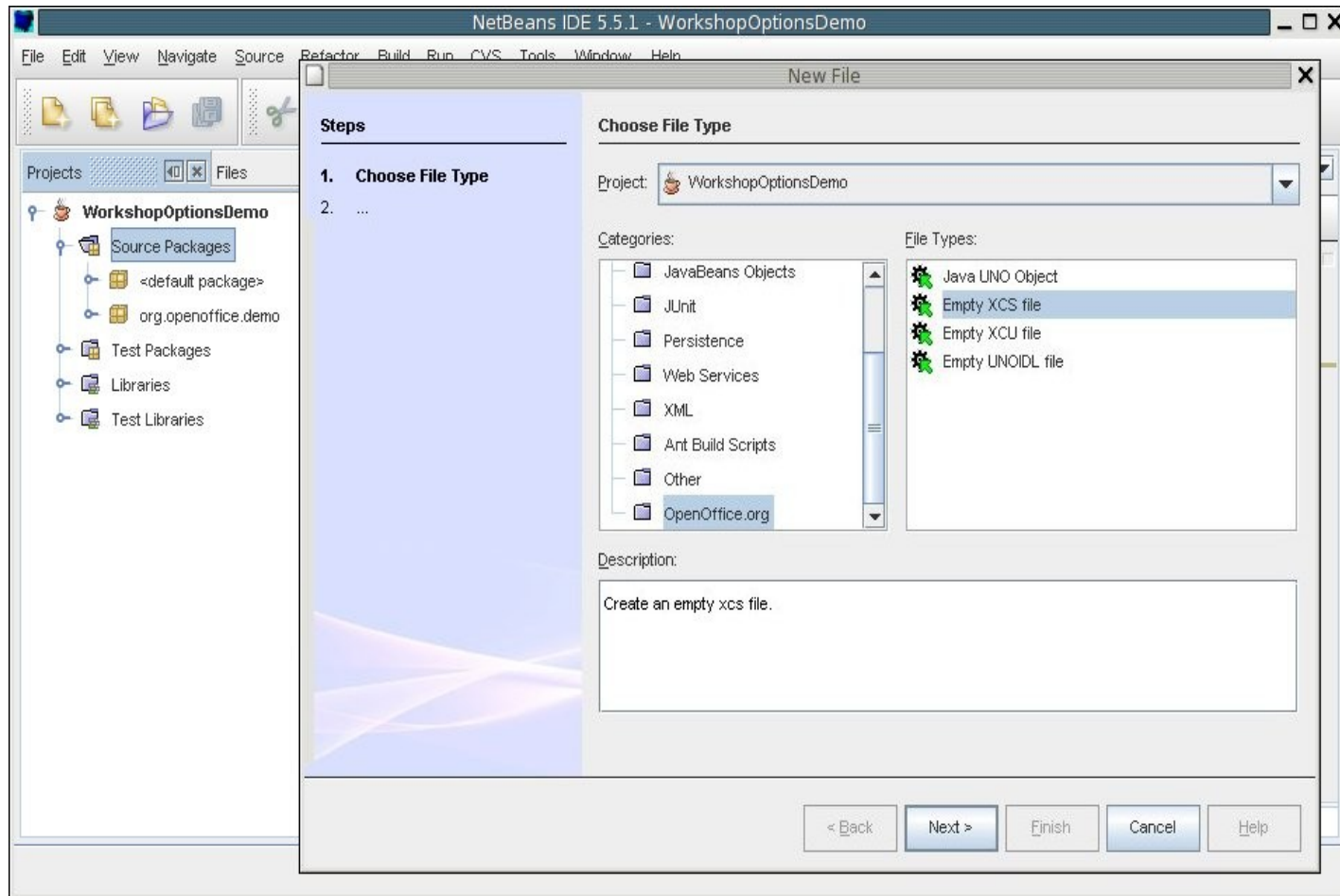
Context in which the command is visible

< Back   Next >   Finish   Cancel   Help



# Demo I: Addon with Option page

- Create a new configuration schema (I)







# Demo I: Addon with Option page

- Create a new configuration schema (II)

New Empty XCS file

**Steps**

1. Choose File Type
2. **Name and Location**

**Name and Location**

File Name: WorkshopOptionsDemo

Project: WorkshopOptionsDemo

Folder: src

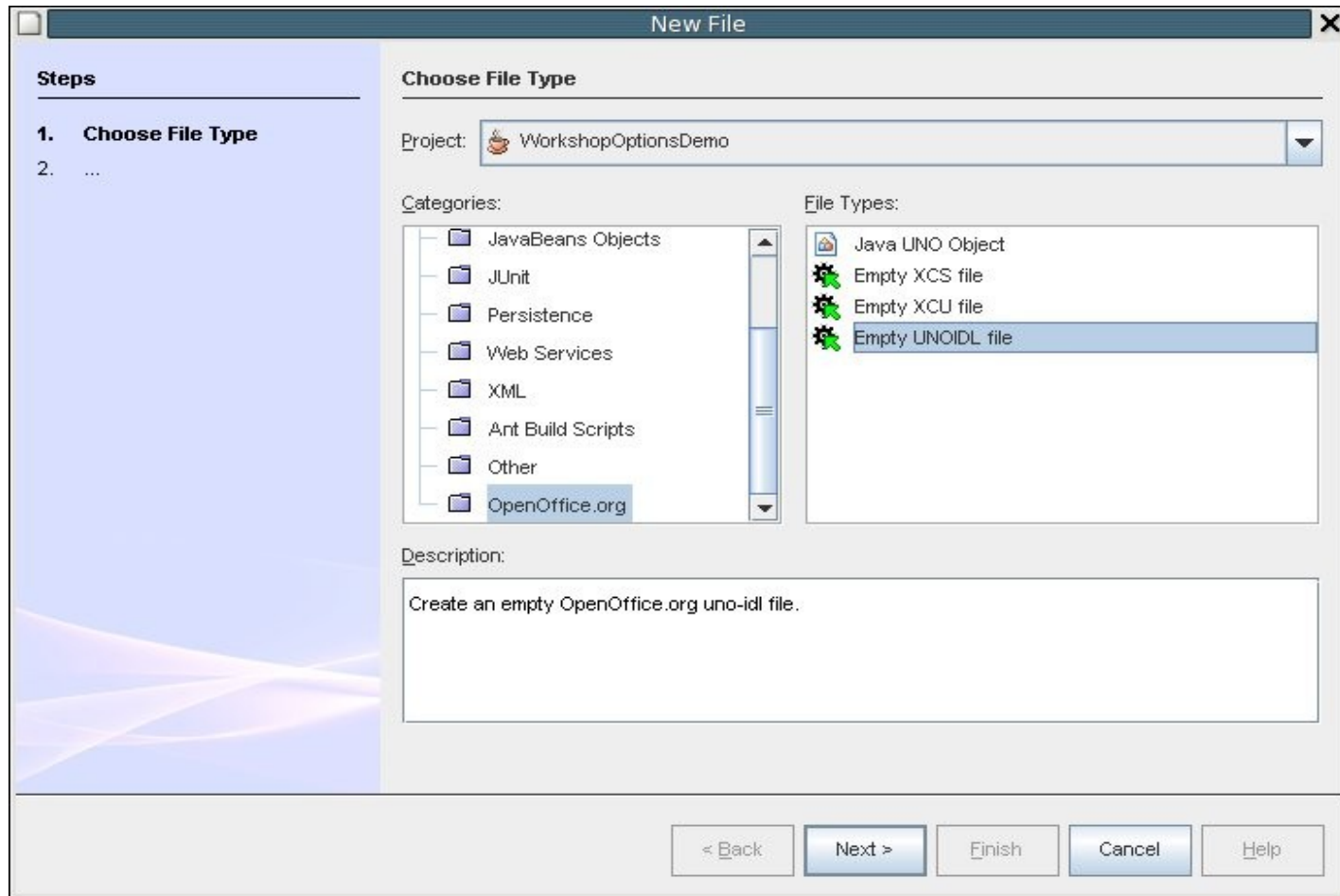
Created File: /export/demo/WorkshopOptionsDemo/src/WorkshopOptionsDemo.xcs

< Back   Next >   Finish   Cancel   Help



# Demo I: Addon with Option page

- Create a new OptionHandler service with implementation (I)





# Demo I: Addon with Option page

- Create a new OptionHandler service with implementation (II)

**Steps**

1. Choose File Type
- 2. Name and Package**
3. UNOIDL File Details

**Name and Package**

Name: DemoOptionsHandler

Data Type: Service

Project: WorkshopOptionsDemo

Location: Source Packages

Package: org.openoffice.demo

Created File: rkshopOptionsDemo/src/org/openoffice/demo/DemoOptionsHandler.idl

Also Create an Implementation Object

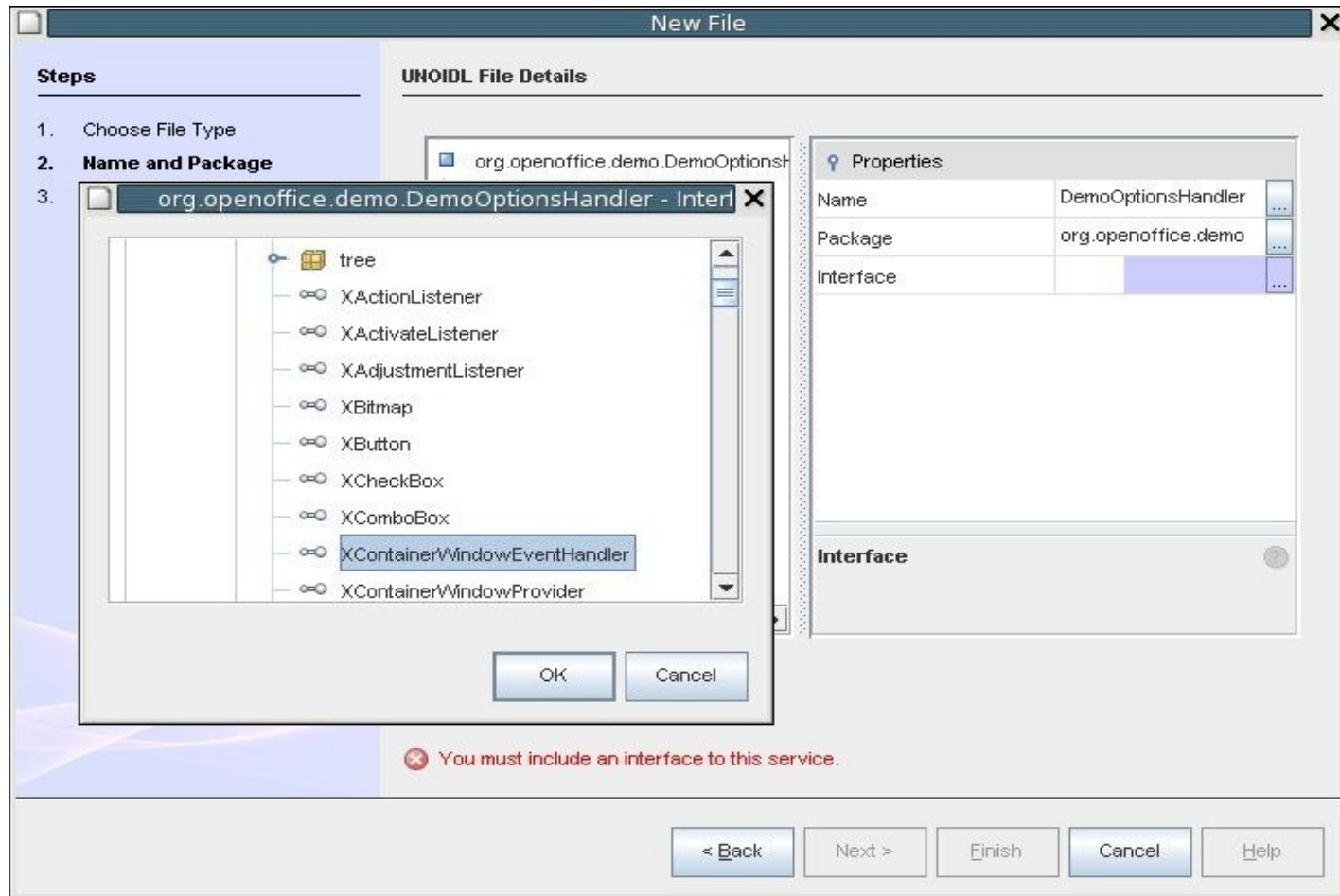
pOptionsDemo/src/org/openoffice/demo/DemoOptionsHandlerImpl.java

< Back   Next >   Finish   Cancel   Help



# Demo I: Addon with Option page

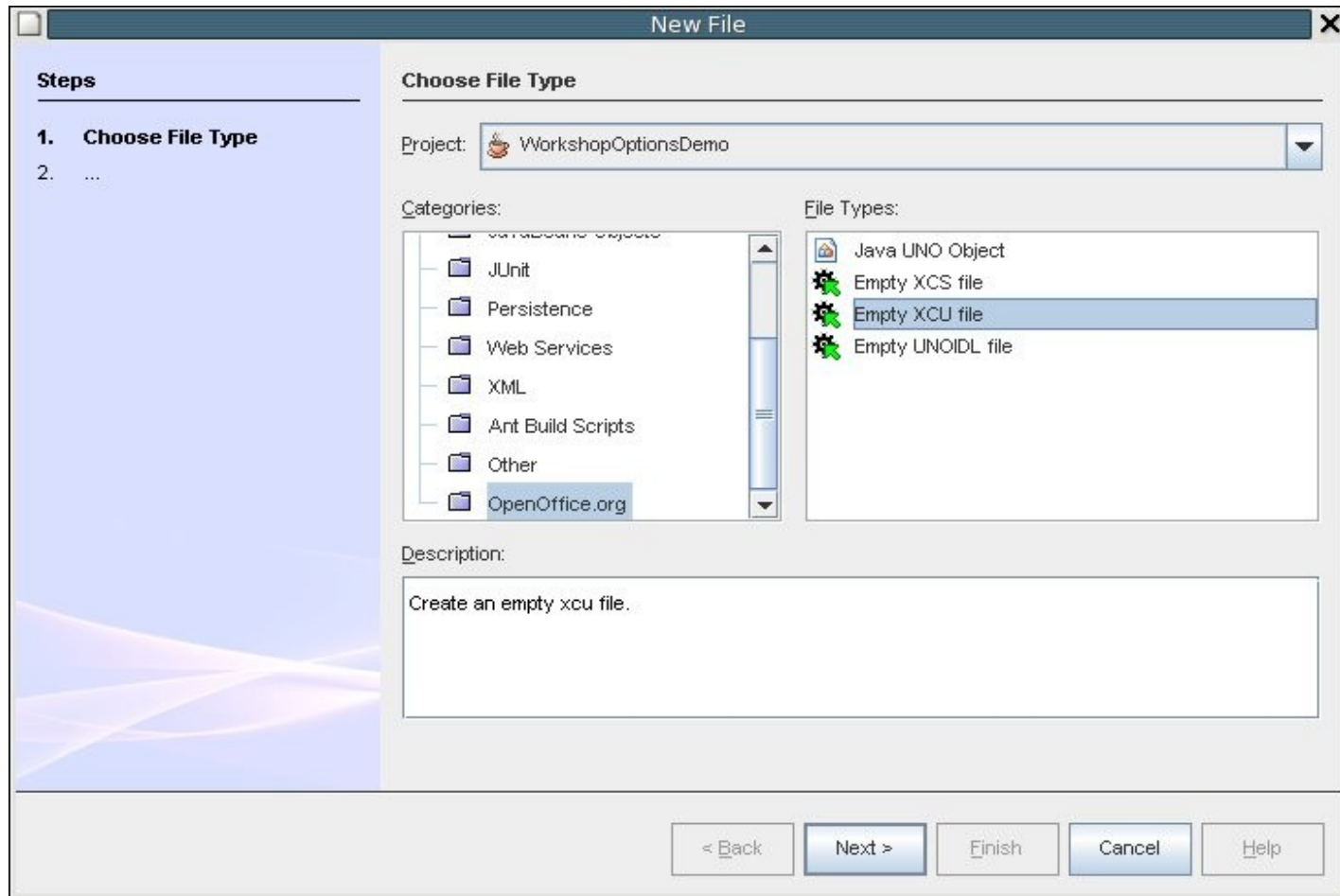
- Create a new OptionHandler service with implementation (III)





# Demo I: Addon with Option page

- Create a new DemoOptionsDialog configuration file (I)





# Demo I: Addon with Option page

- Create a new DemoOptionsDialog configuration file (II)

New Empty XCU file

**Steps**

1. Choose File Type
2. **Name and Location**

**Name and Location**

File Name: DemoOptionsDialog

Project: WorkshopOptionsDemo

Folder: src

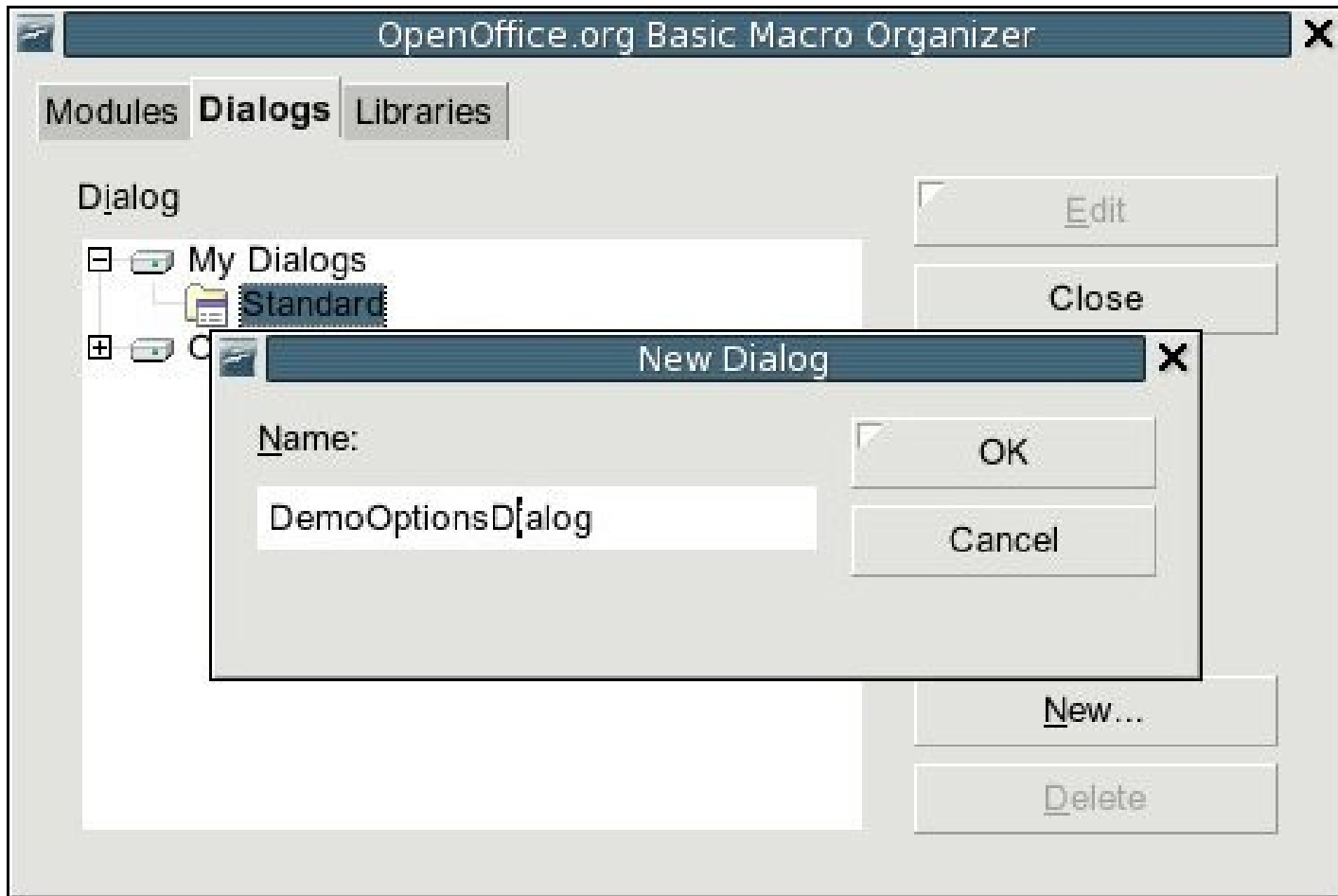
Created File: /export/demo/WorkshopOptionsDemo/src/DemoOptionsDialog.xcu

< Back   Next >   Finish   Cancel   Help



# Demo I: Addon with Option page

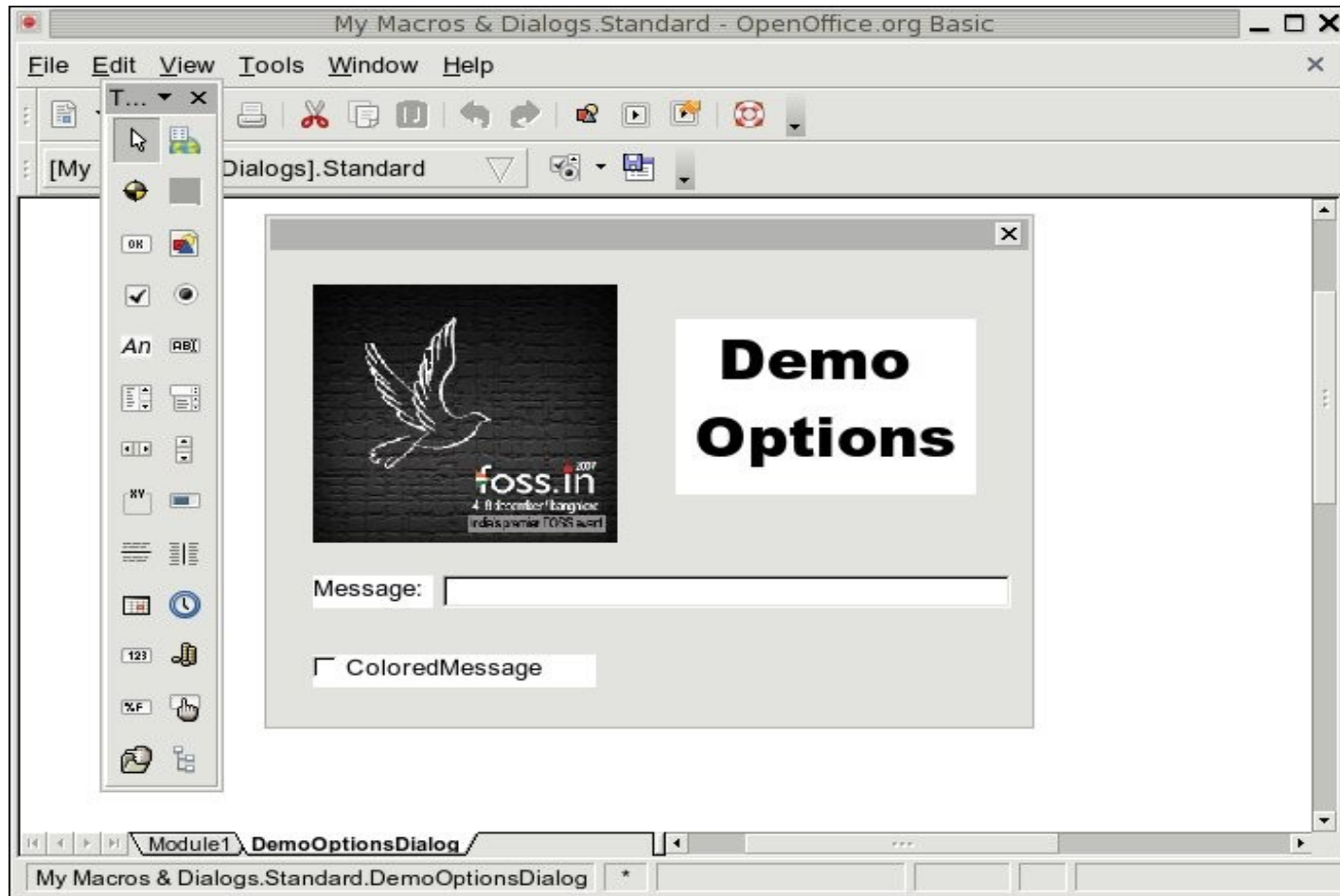
- Create a new DemoOptionsDialog with the Office (I)





# Demo I: Addon with Option page

- Create a new DemoOptionsDialog with the Office (II)

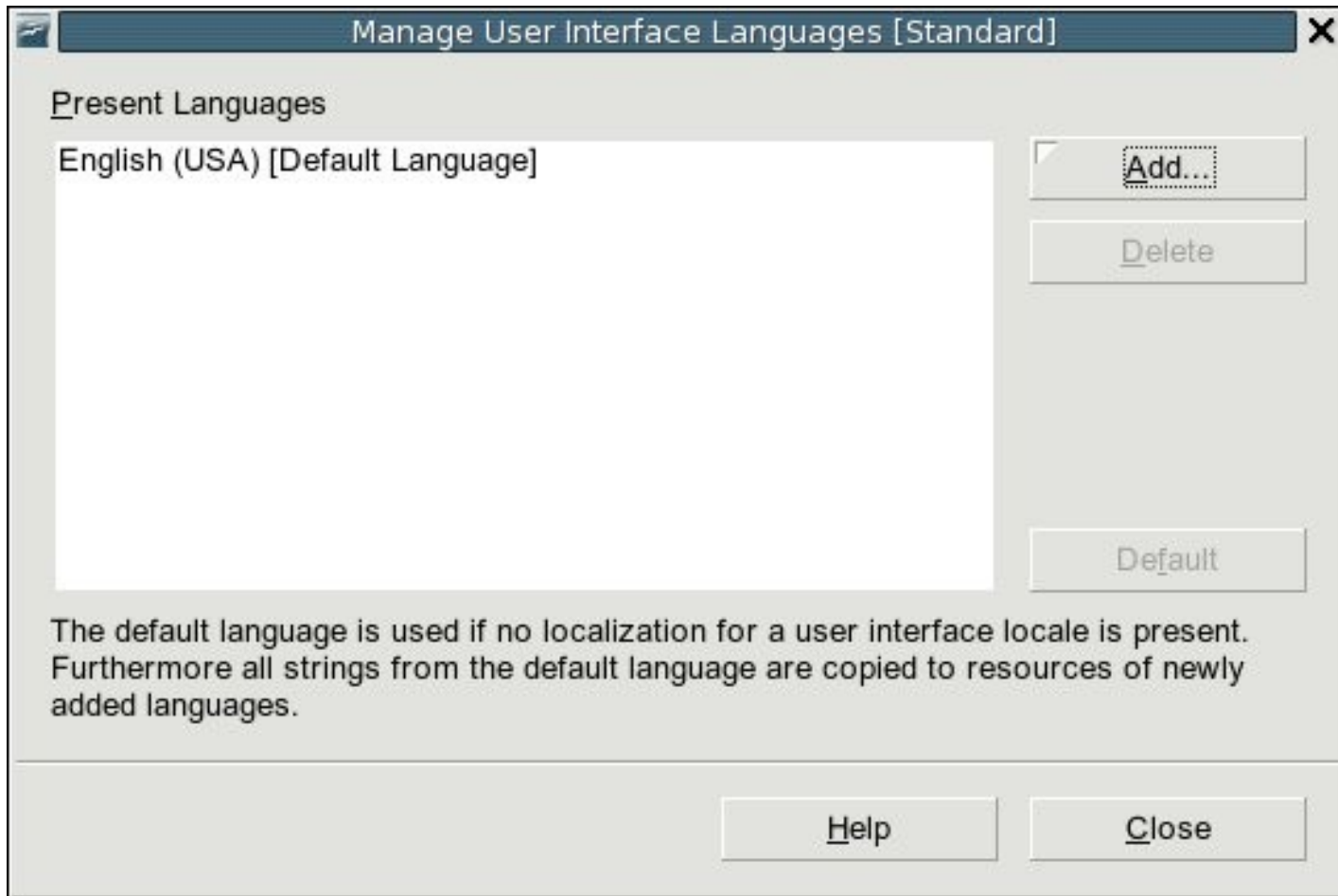






# Demo I: Addon with Option page

- Create a new DemoOptionsDialog with the Office (III)





# Demo I: OOIssue Smart Tag

- Define the configuration schema (WorkshopOptionsDemo.xcs)

```
<?xml version="1.0" encoding="UTF-8"?>
<oor:component-schema xmlns:oor="http://openoffice.org/2001/registry"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  oor:name="WorkshopOptions" oor:package="org.openoffice.workshop.demo"
  xml:lang="en-US">
  <info>
    <author></author>
    <esc>Contains the options data used for the workshop demo extension.</desc>
  </info>
  <component>
    <group oor:name="DemoOptions">
      <info>
        <desc>The data properties for the demo option page.</desc>
      </info>
      <prop oor:name="Message" oor:type="xs:string">
        <value>Hello FOSS.in Conference 2007</value>
      </prop>
      <prop oor:name="ColoredMessage" oor:type="xs:boolean">
        <value>>false</value>
      </prop>
    </group>
  </component>
</oor:component-schema>
```



# Demo I: OOIssue Smart Tag

- Define the configuration for dialog (DemoOptionsDialog.xcu)

```
<?xml version="1.0" encoding="UTF-8"?>
<oor:component-data xmlns:oor="http://openoffice.org/2001/registry"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  oor:name="OptionsDialog" oor:package="org.openoffice.Office">
  <node oor:name="Nodes">
    <node oor:name="Writer" oor:op="fuse">
      <node oor:name="Leaves">
        <node oor:name="org.openoffice.workshop.demo.options" oor:op="fuse">
          <prop oor:name="Id">
            <value>org.openoffice.demo.WorkshopOptionsDemo</value>
          </prop>
          <prop oor:name="Label">
            <value xml:lang="en-US">Workshop Demo</value>
          </prop>
          <prop oor:name="OptionsPage">
            <value>%origin%/dialogs/DemoOptionsDialog.xdl</value>
          </prop>
          <prop oor:name="EventHandlerService">
            <value>org.openoffice.demo.DemoOptionsHandler</value>
          </prop>
        </node>
      </node></node></node>
    </oor:component-data>
```



# Demo I: OOIssue Smart Tag

- Extend the generated WorkshopOptionsDemo Add-On

```

...
private XPropertySet m_xDemoOptions = null;
...
// com.sun.star.frame.XDispatch:
public void dispatch(URL aURL, PropertyValue[] aArguments ) {
    if ( aURL.Protocol.compareTo("org.openoffice.demo.workshopoptionsdemo:") == 0 ) {
        if ( aURL.Path.compareTo("insertMsg") == 0 ) {
            try {
                XTextDocument xDoc = (XTextDocument)UnoRuntime.queryInterface(
                    XTextDocument.class, m_xFrame.getController().getModel());
                XText xText = xDoc.getText();
                XTextCursor xCursor = xText.createTextCursor();
                xCursor.gotoEnd(false);
                xText.insertString(xCursor, readMessage(), true);
                setTextColor(xCursor);
                xCursor.gotoEnd(false);
                xText.insertControlCharacter(xCursor, ControlCharacter.PARAGRAPH_BREAK,
                    false);
            } catch (com.sun.star.uno.Exception e) {
                e.printStackTrace();
            }
        }
    }
}

```



# Demo I: OOIssue Smart Tag

- Extend the generated WorkshopOptionsDemo Add-On

```
private String readMessage() {
    String msg = "";
    if (m_xDemoOptions == null ) initOptionsData();

    try {
        msg = AnyConverter.toString(m_xDemoOptions.getPropertyValue("Message"));
    } catch (com.sun.star.uno.Exception e) { e.printStackTrace(); }
    return msg;
}

private void setTextColor(XTextCursor xCursor) {
    if (m_xDemoOptions == null ) initOptionsData();

    try {
        if (AnyConverter.toBoolean(m_xDemoOptions.getPropertyValue("ColoredMessage"))) {
            XPropertySet xProp = (XPropertySet)UnoRuntime.queryInterface(
                XPropertySet.class, xCursor);
            int c = AnyConverter.toInt(xProp.getPropertyValue("CharColor"));
            xProp.setPropertyValue("CharColor", 0xff0000);
            xCursor.collapseToEnd();
            xProp.setPropertyValue("CharColor", c);
        }
    } catch (com.sun.star.uno.Exception e) { e.printStackTrace(); }
}
```



# Demo I: OOIssue Smart Tag

- Extend the generated WorkshopOptionsDemo Add-On

```
private void initOptionsData() {
    try {
        XMultiServiceFactory xConfig = (XMultiServiceFactory)
            UnoRuntime.queryInterface(XMultiServiceFactory.class,
                m_xContext.getServiceManager().createInstanceWithContext(
                    "com.sun.star.configuration.ConfigurationProvider", m_xContext));

        Object[] args = new Object[1];
        args[0] = new PropertyValue("nodepath", 0,
            "/org.openoffice.workshop.demo.WorkshopOptions/DemoOptions",
            PropertyState.DIRECT_VALUE);

        m_xDemoOptions = (XPropertySet) UnoRuntime.queryInterface(XPropertySet.class,
            xConfig.createInstanceWithArguments(
                "com.sun.star.configuration.ConfigurationAccess", args));
    } catch (com.sun.star.uno.Exception e) {
        e.printStackTrace();
    }
}
```



# Demo I: OOIssue Smart Tag

- Extend the generated DemoOptionsHandler object

```

...
private final XMultiComponentFactory m_xMCF;
private XPropertySet m_xPropOptions = null;
private final String m_resRootURL;
...

public DemoOptionsHandlerImpl( XComponentContext context ) {
    m_xContext = context;
    m_xMCF = m_xContext.getServiceManager();

    XPackageInformationProvider xPkgInfo = PackageInformationProvider.get(m_xContext);
    m_resRootURL = xPkgInfo.getPackageLocation(
        "org.openoffice.demo.WorkshopOptionsDemo") + "/dialogs/";

    XMultiServiceFactory xConfig;
    try {
        xConfig = (XMultiServiceFactory) UnoRuntime.queryInterface(
            XMultiServiceFactory.class, xMCF.createInstanceWithContext(
                "com.sun.star.configuration.ConfigurationProvider", m_xContext));
    } catch (com.sun.star.uno.Exception e) {
        e.printStackTrace();
        return;
    }
    ...

```



# Demo I: OOIssue Smart Tag

- Extend the generated DemoOptionsHandler object

```

...
public DemoOptionsHandlerImpl( XComponentContext context ) {
    ...
    Object[] args = new Object[1];
    args[0] = new PropertyValue("nodepath", 0,
        "/org.openoffice.workshop.demo.WorkshopOptions/DemoOptions",
        PropertyState.DIRECT_VALUE);

    try {
        m_xPropOptions = (XPropertySet) UnoRuntime.queryInterface(XPropertySet.class,
            xConfig.createInstanceWithArguments(
                "com.sun.star.configuration.ConfigurationUpdateAccess", args));
    } catch (com.sun.star.uno.Exception e) {
        e.printStackTrace();
        return;
    }
};

```





# Demo I: OOIssue Smart Tag

- Extend the generated DemoOptionsHandler object

```

...
// com.sun.star.awt.XContainerWindowEventHandler:
public boolean callHandlerMethod(com.sun.star.awt.XWindow xWindow,
    Object aEventObject, String sMethodName)
    throws com.sun.star.lang.WrappedTargetException {

    if (sMethodName.equals("external_event") ) {
        try {
            return handleExternalEvent(xWindow, aEventObject);
        } catch (com.sun.star.uno.RuntimeException re) {
            throw re;
        } catch (com.sun.star.uno.Exception e) {
            throw new WrappedTargetException(sMethodName, this, e);
        }
    }
    return true;
}

public String[] getSupportedMethodNames() {
    return new String[] {"external_event"};
}

```



# Demo I: OOIssue Smart Tag

- Extend the generated DemoOptionsHandler object

```

...
private boolean handleExternalEvent(com.sun.star.awt.XWindow aWindow,
    Object aEventObject)
    throws com.sun.star.uno.Exception {

    try {
        String sMethod = AnyConverter.toString(aEventObject);

        if (sMethod.equals("ok")) {
            saveData(aWindow);
        } else if (sMethod.equals("back") || sMethod.equals("initialize")) {
            loadData(aWindow);
        }
    } catch (com.sun.star.lang.IllegalArgumentException e) {
        throw new com.sun.star.lang.IllegalArgumentException(
            "Method external_event requires a string in the event object argument.",
            this, (short) -1);
    }
    return true;
}

```



# Demo I: OOIssue Smart Tag

- Extend the generated DemoOptionsHandler object

```
private void saveData(com.sun.star.awt.XWindow aWindow)
    throws com.sun.star.lang.IllegalArgumentException, com.sun.star.uno.Exception {

    XControlContainer xContainer = (XControlContainer) UnoRuntime.queryInterface(
        XControlContainer.class, aWindow);

    if (xContainer == null)
        throw new com.sun.star.uno.Exception(
            "Could not get XControlContainer from window.", this);

    XControl xControl = xContainer.getControl("ColoredMessage");
    if (xControl != null) {
        XCheckBox xCheckBox = (XCheckBox)UnoRuntime.queryInterface(
            XCheckBox.class, xControl);

        if (xCheckBox.getState() == 0)
            m_xPropOptions.setPropertyValue("ColoredMessage", new Boolean(false));
        else
            m_xPropOptions.setPropertyValue("ColoredMessage", new Boolean(true));
    }

    ...
}
```



# Demo I: OOIssue Smart Tag

- Extend the generated DemoOptionsHandler object

```
private void saveData(com.sun.star.awt.XWindow aWindow)
    throws com.sun.star.lang.IllegalArgumentException, com.sun.star.uno.Exception {

    ...

    xControl = xContainer.getControl("Message");
    if (xControl != null) {
        XPropertySet xProp = (XPropertySet) UnoRuntime.queryInterface(
            XPropertySet.class, xControl.getModel());

        Object aText = xProp.getPropertyValue("Text");
        m_xPropOptions.setPropertyValue("Message", AnyConverter.toString(aText));
    }

    //Committing the changes will cause or changes to be written to the registry.
    XChangesBatch xUpdateCommit = (XChangesBatch) UnoRuntime.queryInterface(
        XChangesBatch.class, m_xPropOptions);
    xUpdateCommit.commitChanges();
}
```



# Demo I: OOIssue Smart Tag

- Extend the generated DemoOptionsHandler object

```
private void loadData(com.sun.star.awt.XWindow aWindow)
    throws com.sun.star.uno.Exception {

    XControlContainer xContainer = (XControlContainer) UnoRuntime.queryInterface(
        XControlContainer.class, aWindow);

    if (xContainer == null)
        throw new com.sun.star.uno.Exception(
            "Could not get XControlContainer from window.", this);

    // init the image control, just a workaround
    XControl xControl = xContainer.getControl("LogoImage");
    if (xControl != null) {
        XPropertySet xProp = (XPropertySet) UnoRuntime.queryInterface(
            XPropertySet.class, xControl.getModel());

        xProp.setPropertyValue("ImageURL", m_resRootURL + "fossinlogo.jpg");
    }

    ...
}
```



# Demo I: OOIssue Smart Tag

- Extend the generated DemoOptionsHandler object

```
private void loadData(com.sun.star.awt.XWindow aWindow)
    throws com.sun.star.uno.Exception {

    ...

    xControl = xContainer.getControl("ColoredMessage");
    if (xControl != null) {
        XCheckBox xCheckBox = (XCheckBox)UnoRuntime.queryInterface(
            XCheckBox.class, xControl);
        Object aValue = m_xPropOptions.getPropertyValue("ColoredMessage");

        if (AnyConverter.toBoolean(aValue))
            xCheckBox.setState((short)1);
        else
            xCheckBox.setState((short)0);
    }

    ...
}
```



# Demo I: OOIssue Smart Tag

- Extend the generated DemoOptionsHandler object

```
private void loadData(com.sun.star.awt.XWindow aWindow)
    throws com.sun.star.uno.Exception {

    ...

    xControl = xContainer.getControl("Message");
    if (xControl != null) {
        Object aValue = m_xPropOptions.getPropertyValue("Message");

        XPropertySet xProp = (XPropertySet) UnoRuntime.queryInterface(
            XPropertySet.class, xControl.getModel());

        xProp.setPropertyValue("Text", aValue);
    }
}
```



# Demo

**Step By Step example**

**WorkshopDemoAddon  
Extensions with own an  
Option page**





# More information

- API project
  - > home page: [api.openoffice.org](http://api.openoffice.org) mailing list: [dev@api.openoffice.org](mailto:dev@api.openoffice.org)
  - > IRC (freenode): #ooo-api for all API relevant topics
- Extensions project
  - > home page: [extensions.openoffice.org](http://extensions.openoffice.org)
  - > mailing list: [dev@extensions.openoffice.org](mailto:dev@extensions.openoffice.org)
  - > IRC (freenode): #ooo-ext for general extension topics
- OpenOffice.org Wiki
  - > OpenOffice.org API plugin for NetBeans
    - [http://wiki.services.openoffice.org/wiki/OpenOffice\\_NetBeans\\_Integration](http://wiki.services.openoffice.org/wiki/OpenOffice_NetBeans_Integration)



# Q & A



# OpenOffice.org Extensions development in Java with NetBeans in practise

Jürgen Schmidt

[juergen.schmidt@sun.com](mailto:juergen.schmidt@sun.com)