



OpenOffice.org Extensions With NetBeansTM Software

Juergen Schmidt

Senior Software Engineer
Sun Microsystems
www.sun.com

TS-7557

Goal of My Talk

What My Audience Will Gain

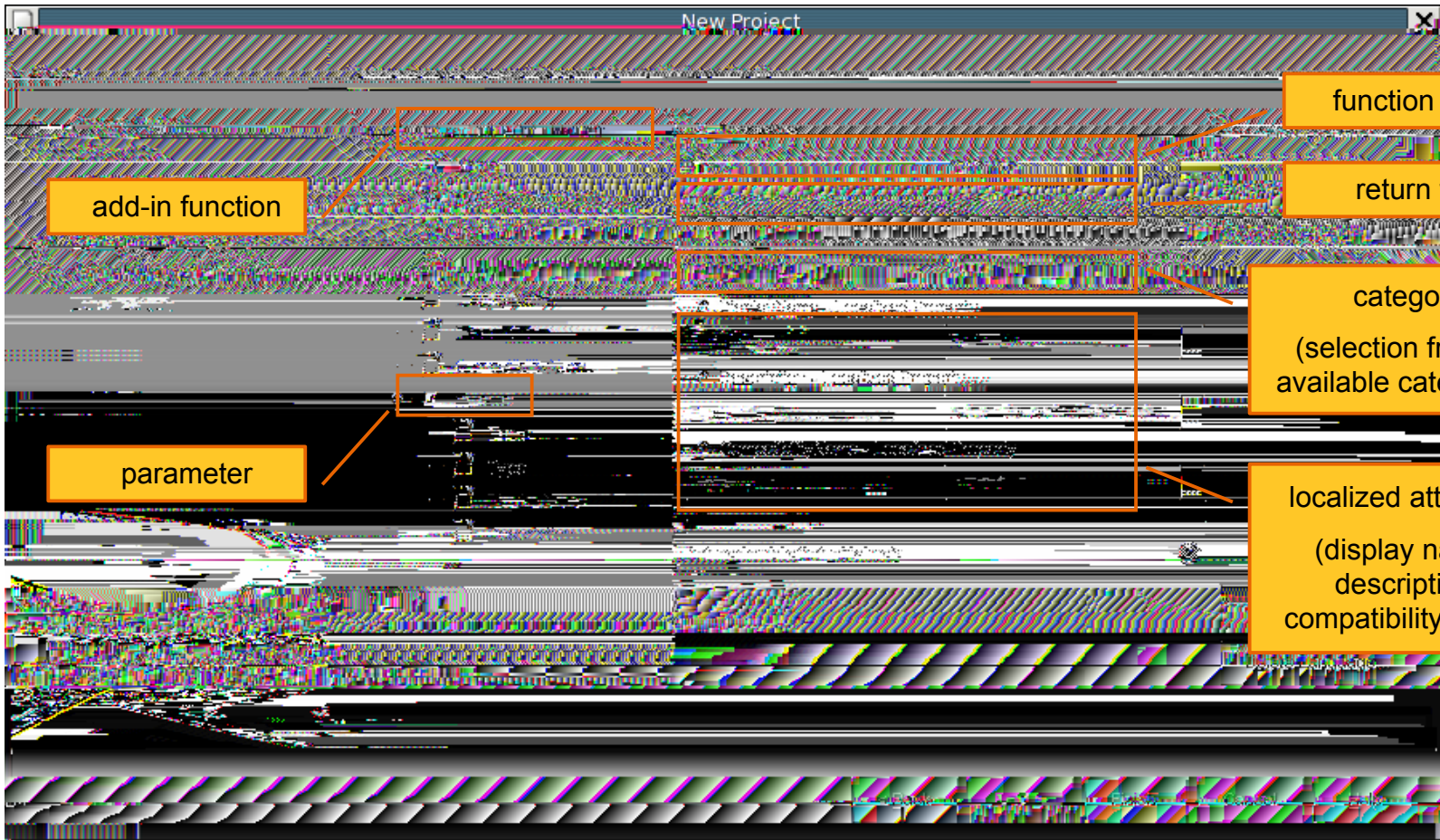
Learn how to extend or integrate into OpenOffice.org, the most popular ODF manipulating office suite, with Java™ technology and NetBeans software. Learn, for example, how to create a built-in function for the Calc application in five minutes.

About the Speaker

- Senior Software Engineer at Sun Microsystems
- Main focus: OpenOffice.org programmability
 - API, UNO, OpenOffice.org SDK, Extensions
 - OpenOffice.org Extensions NetBeans software plugin, ODF Toolkit
- OpenOffice.org/StarOffice™ software developer since 1997
- OpenOffice.org API project lead
- Co-lead of the Extensions and ODFToolkit projects

Demo: Calc Built-In Function

To serve the statement to create a new one in 5

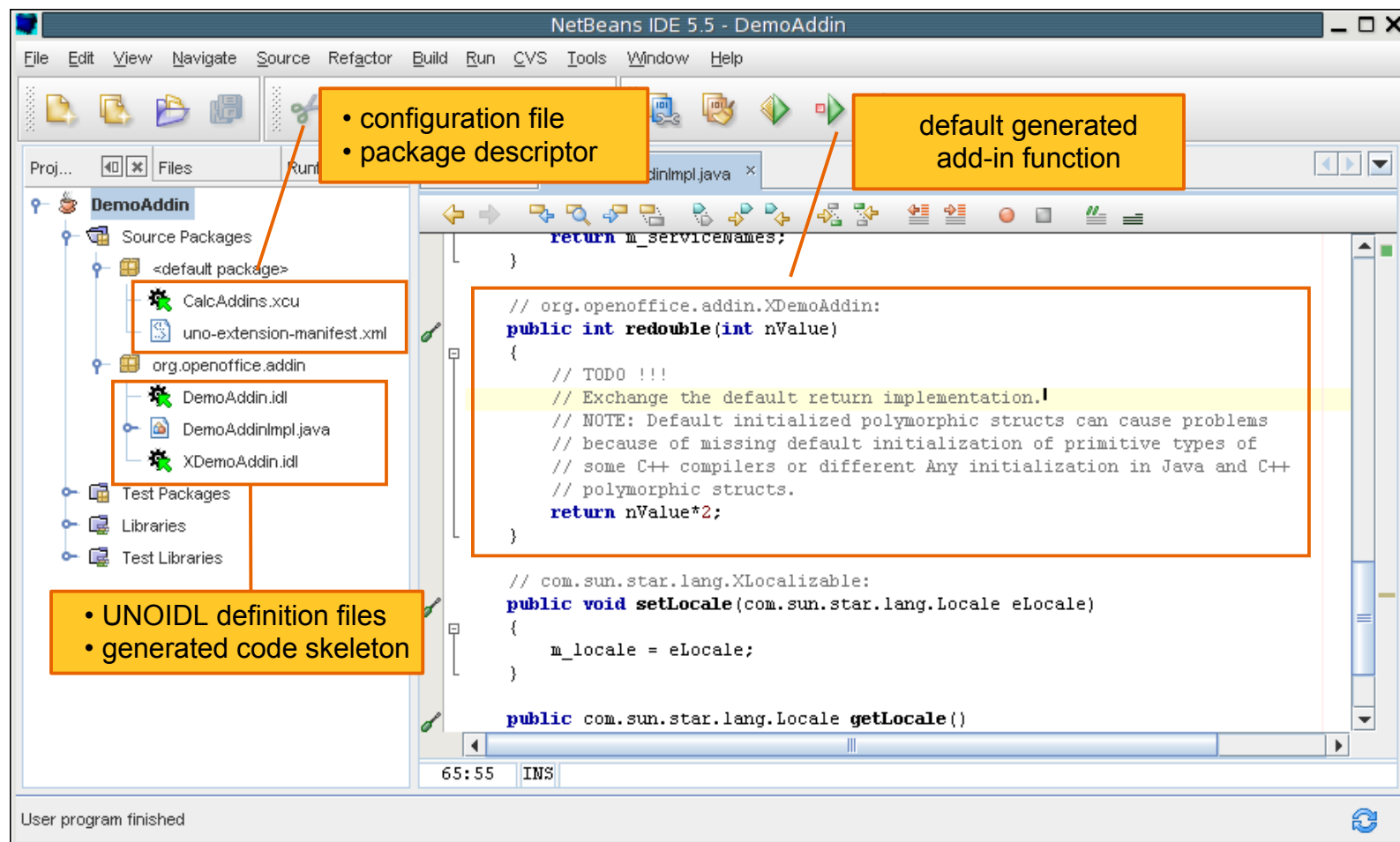


The screenshot shows the 'New Project' dialog box with several fields and buttons. Annotations point to the following elements:

- add-in function**: Points to the 'Add-in function' button.
- function name**: Points to the 'Function name' text field.
- return type**: Points to the 'Return type' dropdown menu.
- category** (selection from all available categories): Points to the 'Category' dropdown menu.
- parameter**: Points to the 'Parameter' text field.
- localized attributes** (display name, description, compatibility name): Points to the 'Localized attributes' text area.

Demo: Calc Built-In Function

To serve the statement to create a new one in 5



The screenshot shows the NetBeans IDE 5.5 interface with the 'DemoAddin' project open. The left sidebar displays the project structure, and the main editor shows the implementation of the 'redouble' function in 'XDemoAddin.idl'.

Project Structure (Left Sidebar):

- Source Packages
 - <default package>
 - CalcAddins.xcu (configuration file)
 - uno-extension-manifest.xml (package descriptor)
 - org.openoffice.addin
 - DemoAddin.idl (UNOIDL definition file)
 - DemoAddinImpl.java (generated code skeleton)
 - XDemoAddin.idl (UNOIDL definition file)
- Test Packages
- Libraries
- Test Libraries

Code Editor (Main Window):

```

return m_servicenames;
}

// org.openoffice.addin.XDemoAddin:
public int redouble(int nValue)
{
    // TODO !!!
    // Exchange the default return implementation.
    // NOTE: Default initialized polymorphic structs can cause problems
    // because of missing default initialization of primitive types of
    // some C++ compilers or different Any initialization in Java and C++
    // polymorphic structs.
    return nValue*2;
}

// com.sun.star.lang.XLocalizable:
public void setLocale(com.sun.star.lang.Locale eLocale)
{
    m_locale = eLocale;
}

public com.sun.star.lang.Locale getLocale()
  
```

Annotations:

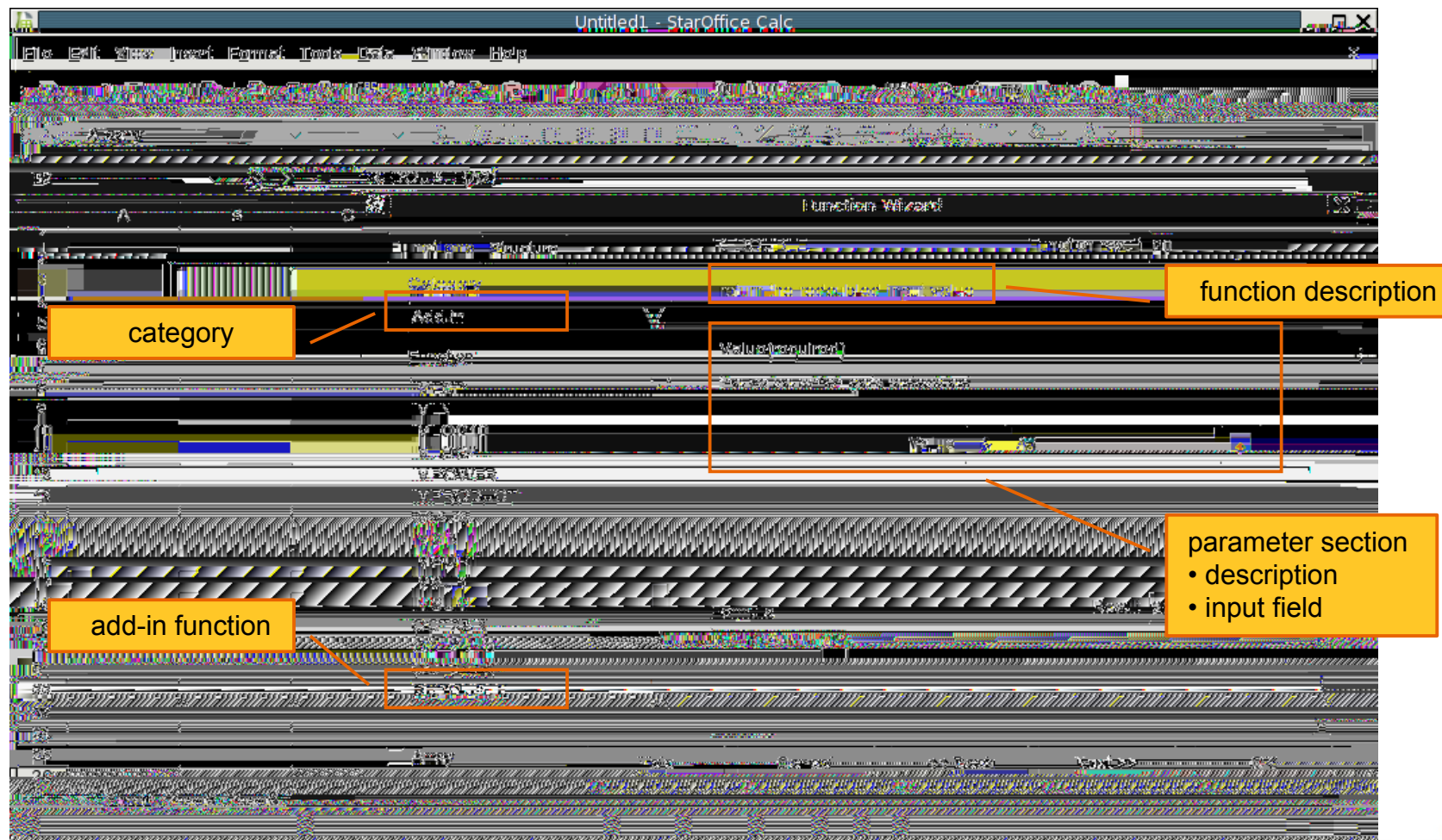
- configuration file** and **package descriptor**: Point to 'CalcAddins.xcu' and 'uno-extension-manifest.xml' respectively.
- default generated add-in function**: Points to the 'redouble' function implementation.
- UNOIDL definition files** and **generated code skeleton**: Point to 'DemoAddin.idl' and 'XDemoAddin.idl' respectively.

65:55 INS

User program finished

Demo: Calc Built-In Function

To serve the statement to create a new one in 5





DEMO

To serve the statement to create a Calc built-in function in 5 minutes

Please check your watches!

Agenda

Motivation for OpenOffice.org Extensions

OpenOffice.org Programmability At-a-Glance

Extensions With NetBeans Software (Demo)

Java Platform Macros With NetBeans Software (Demo)

Summary

Q&A

Motivation for OpenOffice.org Extensions

- Growing popularity of ODF worldwide
 - Standardized file format
 - ODF = Open Document Format for Office Applications
 - OASIS and ISO/IEC 26300
 - Adoption of ODF in more and more public administrations
- Growing popularity of OpenOffice.org
 - Most popular ODF manipulating office suite
 - > 100 Million downloads
 - Multi platform support
 - Solaris™ Operating System (Solaris OS), Linux, Windows, Mac OS, ...

Motivation for OpenOffice.org Extensions

- Demand for
 - Customization of OpenOffice.org
 - User interface changes
 - Exchange, intercept commands
 - Extending OpenOffice.org with new functionality
 - Calc Add-ins, Add-ons, Embedded Java objects, ...
 - Integration in existing workflows or other applications
 - e.g. OpenOffice.org Bean
 - Create, change, convert ODF documents
- Community building
 - Lower the entrance barrier for developers

Agenda

Motivation for OpenOffice.org Extensions

OpenOffice.org Programmability At-a-Glance

Extensions With NetBeans Software (Demo)

Java Platform Macros With NetBeans Software (Demo)

Summary

Q&A

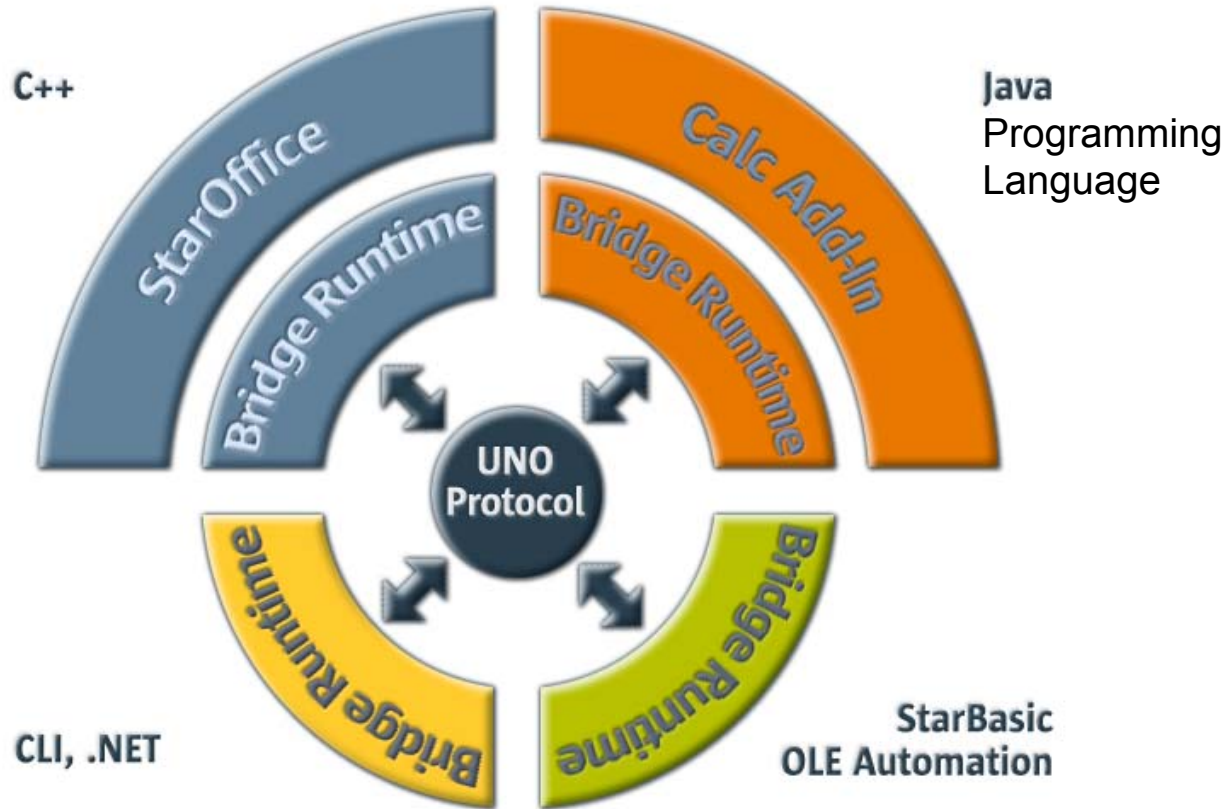
OpenOffice Programmability At-a-Glance

Universal Network Objects (UNO)

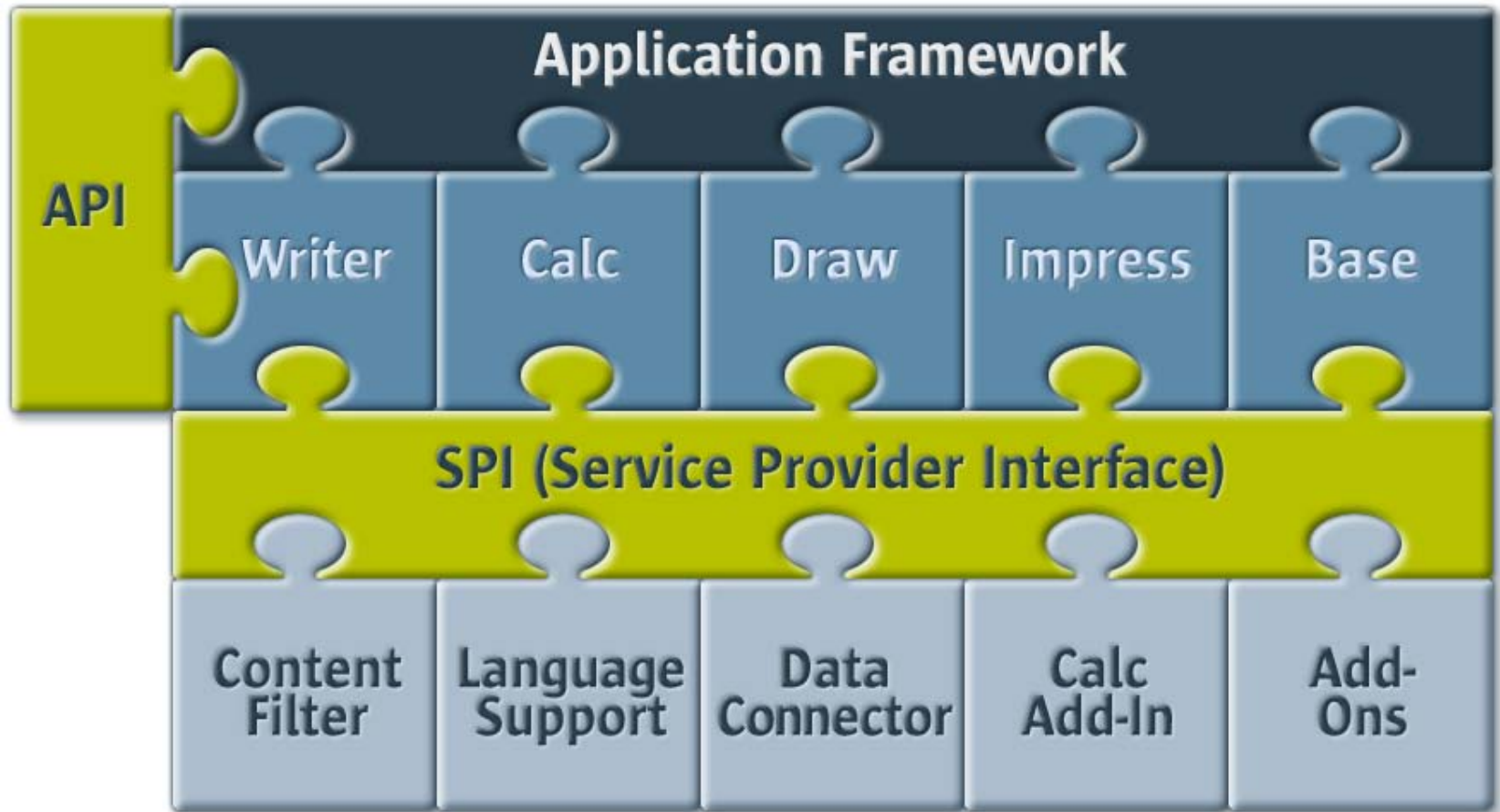
- Component technology \approx RMI/DCOM, Corba
- Language independent
 - API defined in UNOIDL
 - Multi language support (Java programming language, C++, StarBasic, CLI languages, Python)
- API calls work in-process, inter-process or remotely
- Remote transparency
- Can be used without the office
 - URE = UNO Runtime Environment

OpenOffice Programmability At-a-Glance

UNO gets over boundaries



OpenOffice Programmability At-a-Glance



Agenda

Motivation for OpenOffice.org Extensions

OpenOffice.org Programmability At-a-Glance

Extensions With NetBeans Software (Demo)

Java Platform Macros With NetBeans Software (Demo)

Summary

Q&A

Extensions With NetBeans Software

OpenOffice.org API plugin for NetBeans software

- 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

Extensions With NetBeans Software

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

Extensions With NetBeans Software

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

Extensions With NetBeans Software

Calc add-in project wizard

- 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

Extensions With NetBeans Software

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`

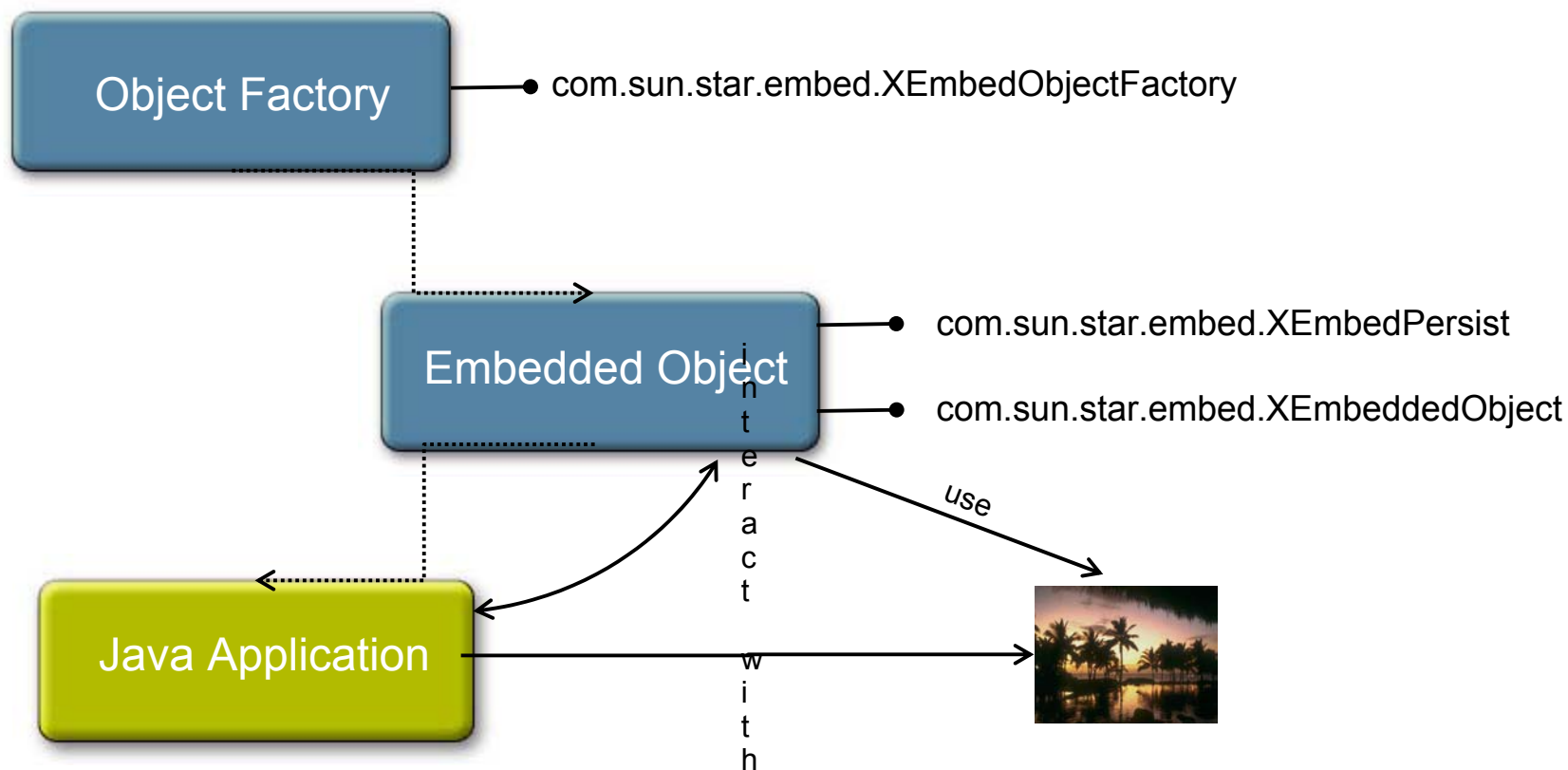
Extensions With NetBeans Software

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

Extensions With NetBeans Software

Example: Java technology Embedded Object





DEMO

OpenOffice.org component projects

Add-on UNO component—
Java technology embedded object

Agenda

Motivation for OpenOffice.org Extensions

OpenOffice.org Programmability At-a-Glance

Extensions With NetBeans Software (Demo)

Java Platform Macros With NetBeans Software (Demo)

Summary

Q&A

Java Platform Macros With NetBeans Software

- Alternative to StarBasic
- Address more experienced developers
 - Object-oriented programming
- Make use of available Java APIs
 - Combining Office APIs with the Java technology world
- Make use of modern IDE features
 - Code completion, debug support

Java Platform Macros With NetBeans Software

- Characteristics of Java platform macro functions
 - `public void function(XScriptContext, Object[])`
 - `com.sun.star.script.provider.XScriptContext`
 - Entry point for all scripts
 - Access to Document, Desktop, Component context
 - `Object[]`
 - Provide access to more script parameters
 - e.g. event objects

Java Platform Macro Sample

```
class Hello {
    public void helloJavaOne(XScriptContext xContext,
                             Object[] params)
    {
        // get document from scripting context and
        // query for a text document
        XTextDocument xDocument = (XTextDocument)
            UnoRuntime.queryInterface(XTextDocument.class,
                                     xContext.getDocument());
        if (xDocument != null) {
            XText xText = xDcoument.getText();
            // goto to the end of the text
            XTextRange xTextRange = xText.getEnd();
            xTextRange.setString("Hello JavaOne");
        }
    }
}
```

Java Platform Macros With NetBeans Software

Support in NetBeans software

- New OpenOffice.org Scripting Project type
- Import/export of Java platform Macro Libraries
 - From/into the user or share layer
 - From/into open documents
- Debug Support
- Export as office extension package (.oxt)



DEMO

OpenOffice.org Scripting Project

Working with Java Platform Macros in
NetBeans Software

Summary

- OpenOffice.org is programmable in Java technology
- Growing tools support
 - Lower entry and fast feeling of success
 - Automation of recurring tasks
 - Speed up development
 - Reduce development costs
- SPI's for specific functional areas
- Office functionality usable in own applications
- Java platform macros for automation of workflows

For More Information

- Hands-on lab 9510 (on DVD), OpenOffice.org Extensions with NetBeans Software
- API project
 - home page: api.openoffice.org
 - mailing list: dev@api.openoffice.org
 - IRC (freenode): #ooo-api for all API relevant topics
- Extensions project
 - home page: extensions.openoffice.org
 - mailing list: dev@extensions.openoffice.org
 - IRC (freenode): #ooo-ext for general extension topics
- http://wiki.services.openoffice.org/wiki/OpenOffice_NetBeans_Integration



Q&A

juergen.schmidt@sun.com

<code/>



OpenOffice.org Extensions With NetBeansTM Software

Juergen Schmidt

Senior Software Engineer
Sun Microsystems
www.sun.com

TS-7557