

# How To... Reuse Business Objects and Override Operations of a Business Object

Applicable Releases:

SAP NetWeaver Composition Environment 7.1

Topic Area:

Development and Composition

Capability:

Composition

Version 1.0

March 2008

© Copyright 2008 SAP AG. All rights reserved.

No part of this publication may be reproduced or transmitted in any form or for any purpose without the express permission of SAP AG. The information contained herein may be changed without prior notice.

Some software products marketed by SAP AG and its distributors contain proprietary software components of other software vendors.

Microsoft, Windows, Outlook, and PowerPoint are registered trademarks of Microsoft Corporation.

IBM, DB2, DB2 Universal Database, OS/2, Parallel Sysplex, MVS/ESA, AIX, S/390, AS/400, OS/390, OS/400, iSeries, pSeries, xSeries, zSeries, z/OS, AFP, Intelligent Miner, WebSphere, Netfinity, Tivoli, Informix, i5/OS, POWER, POWER5, OpenPower and PowerPC are trademarks or registered trademarks of IBM Corporation.

Adobe, the Adobe logo, Acrobat, PostScript, and Reader are either trademarks or registered trademarks of Adobe Systems Incorporated in the United States and/or other countries.

Oracle is a registered trademark of Oracle Corporation.

UNIX, X/Open, OSF/1, and Motif are registered trademarks of the Open Group.

Citrix, ICA, Program Neighborhood, MetaFrame, WinFrame, VideoFrame, and MultiWin are trademarks or registered trademarks of Citrix Systems, Inc.

HTML, XML, XHTML and W3C are trademarks or registered trademarks of W3C®, World Wide Web Consortium, Massachusetts Institute of Technology.

Java is a registered trademark of Sun Microsystems, Inc.

JavaScript is a registered trademark of Sun Microsystems, Inc., used under license for technology invented and implemented by Netscape.

MaxDB is a trademark of MySQL AB, Sweden.

SAP, R/3, mySAP, mySAP.com, xApps, xApp, SAP NetWeaver, and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP AG in Germany and in several other countries all over the world. All other product and service names mentioned are the trademarks of their respective companies. Data contained in this document serves informational purposes only. National product specifications may vary.

These materials are subject to change without notice.

These materials are provided by SAP AG and its affiliated companies ("SAP Group") for informational purposes only, without representation or warranty of any kind, and SAP Group shall not be liable for errors or omissions with respect to the materials. The only warranties for SAP Group products and services are those that are set forth in the express warranty statements accompanying such products and services, if any. Nothing herein should be construed as constituting an additional warranty.

These materials are provided "as is" without a warranty of any kind, either express or implied, including but not limited to, the implied warranties of merchantability, fitness for a particular purpose, or non-infringement.

SAP shall not be liable for damages of any kind including without limitation direct, special, indirect, or consequential damages that may result from the use of these materials.

SAP does not warrant the accuracy or completeness of the information, text, graphics, links or other items contained within these materials. SAP has no control over the information that you may access through the use of hot links contained in these materials and does not endorse your use of third party web pages nor provide any warranty whatsoever relating to third party web pages.

SAP NetWeaver "How-to" Guides are intended to simplify the product implementation. While specific product features and procedures typically are explained in a practical business context, it is not implied that those features and procedures are the only approach in solving a specific business problem using SAP NetWeaver. Should you wish to receive additional information, clarification or support, please refer to SAP Consulting.

Any software coding and/or code lines / strings ("Code") included in this documentation are only examples and are not intended to be used in a productive system environment. The Code is only intended better explain and visualize the syntax and phrasing rules of certain coding. SAP does not warrant the correctness and completeness of the Code given herein, and SAP shall not be liable for errors or damages caused by the usage of the Code, except if such damages were caused by SAP intentionally or grossly negligent.

#### Disclaimer

Some components of this product are based on Java™. Any code change in these components may cause unpredictable and severe malfunctions and is therefore expressly prohibited, as is any decompilation of these components.

Any Java™ Source Code delivered with this product is only to be used by SAP's Support Services and may not be modified or altered in any way.

# Document History

<b>Document Version</b>	<b>Description</b>
1.00	First official release of this guide

## Typographic Conventions

Type Style	Description
<i>Example Text</i>	Words or characters quoted from the screen. These include field names, screen titles, pushbuttons labels, menu names, menu paths, and menu options.  Cross-references to other documentation
<b>Example text</b>	Emphasized words or phrases in body text, graphic titles, and table titles
Example text	File and directory names and their paths, messages, names of variables and parameters, source text, and names of installation, upgrade and database tools.
<b>Example text</b>	User entry texts. These are words or characters that you enter in the system exactly as they appear in the documentation.
<Example text>	Variable user entry. Angle brackets indicate that you replace these words and characters with appropriate entries to make entries in the system.
EXAMPLE TEXT	Keys on the keyboard, for example, F2 or ENTER.

## Icons





Icon	Description
	Caution
	Note or Important
	Example
	Recommendation or Tip

Table of Contents

- 1. **Business Scenario**..... 1
- 2. **Prerequisites** ..... 1
- 3. **Step-by-Step Procedure**..... 2
  - 3.1 Build the first project (Project A) which contains the Business Object ..... 2
  - 3.2 Create the Business Object..... 4
  - 3.3 Create the second project (Project B) ..... 7
  - 3.4 Overriding the Business Object Operations ..... 11

## 1. Business Scenario

Often times when creating application services you will find that the business objects that you need have already been created but are in a different project. With SAP NetWeaver Composition Environment 7.1 it is now possible to reuse business objects that are contained in a different project into an application service. The first part of this guide provides you with details on how to achieve this.

In the second part of this guide we demonstrate how to override the Business Objects default methods. This is also a new feature in the current release of SAP NetWeaver Composition Environment 7.1.

## 2. Prerequisites

Before you start with this tutorial you should have installed the following Software:

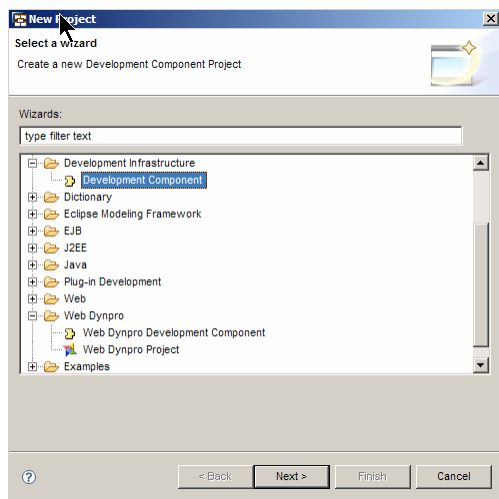
- SAP NetWeaver Developer Studio SAP NetWeaver Composition Environment 7.1 SP01 or higher.
- SAP NetWeaver Composition Environment 7.1 SP01 Application Server.

## 3. Step-by-Step Procedure

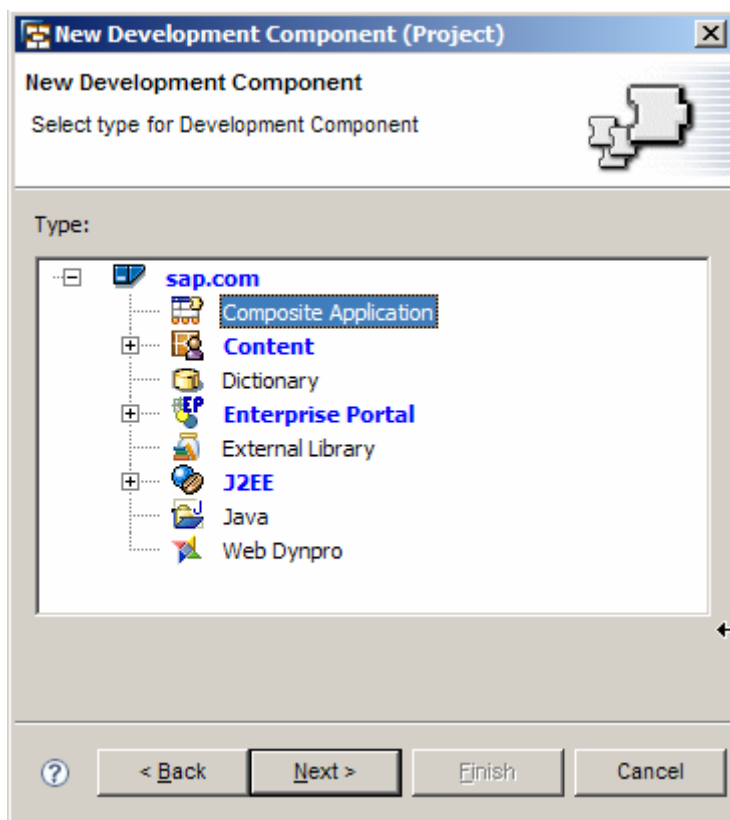
### 3.1 Build the first project (Project A) which contains the Business Object

Launch the NW Developer Studio, and create a new project by choosing *File* → *New* → *Project* from the menu.

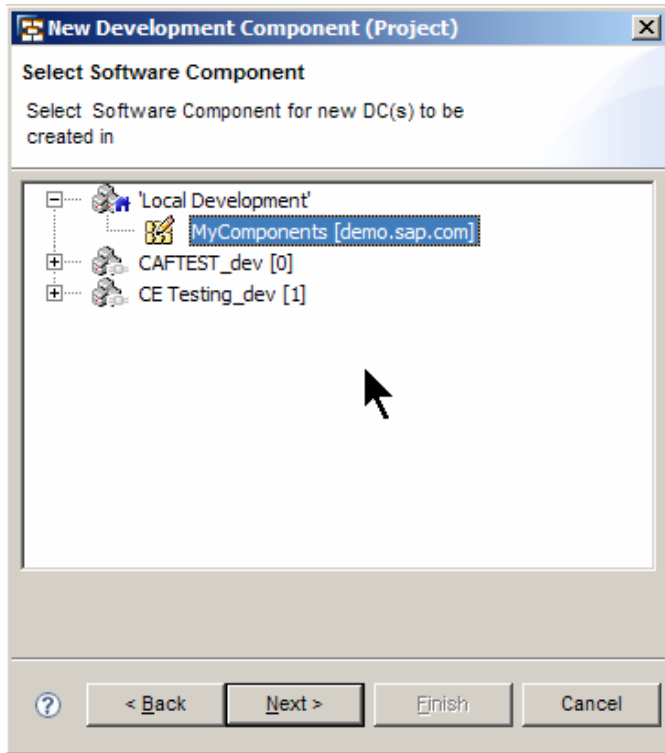
Choose Development Component and click *Next*.



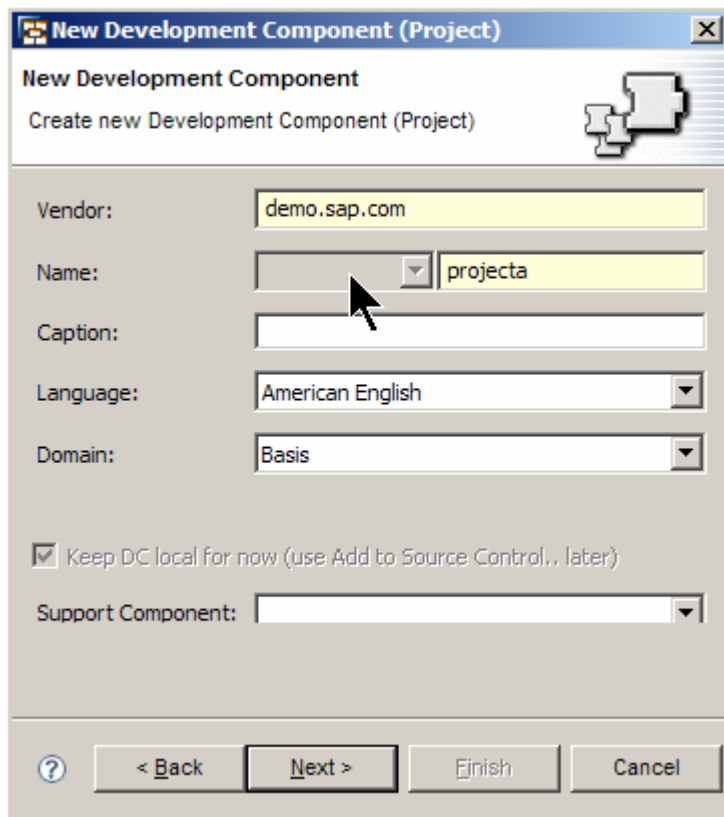
Choose *Composite Application* as the DC type, and click *Next*.



Place the Component in the *Local Development* Configuration. In case you are using NWDI, you can place this component in one of your tracks for CE.

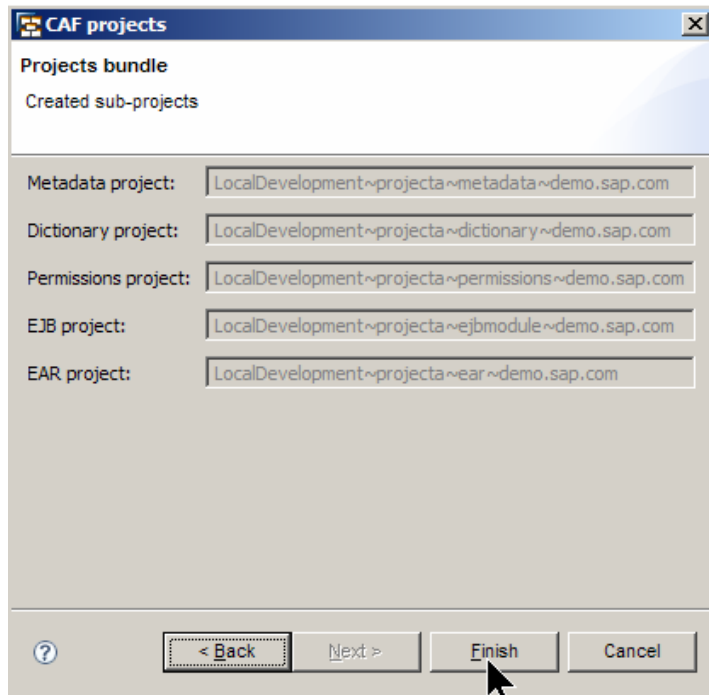


Give the project a name, and click *Next*. Here, we chose **projecta**.





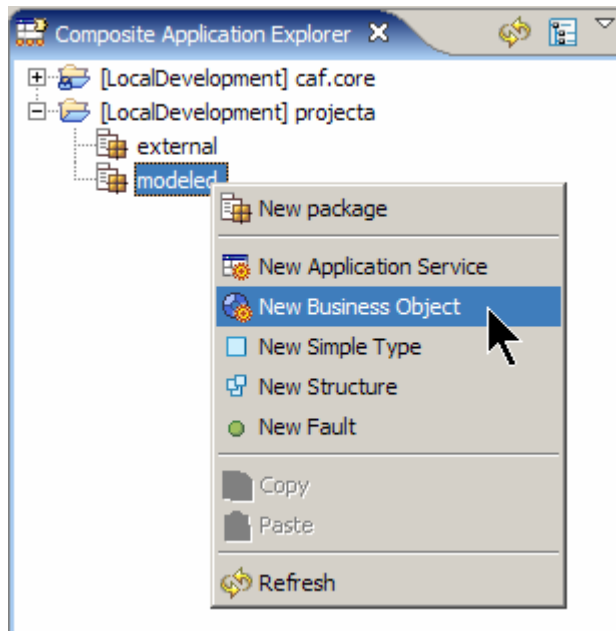
Click *Finish*. This should create all the corresponding projects and automatically opens the Composition Perspective.



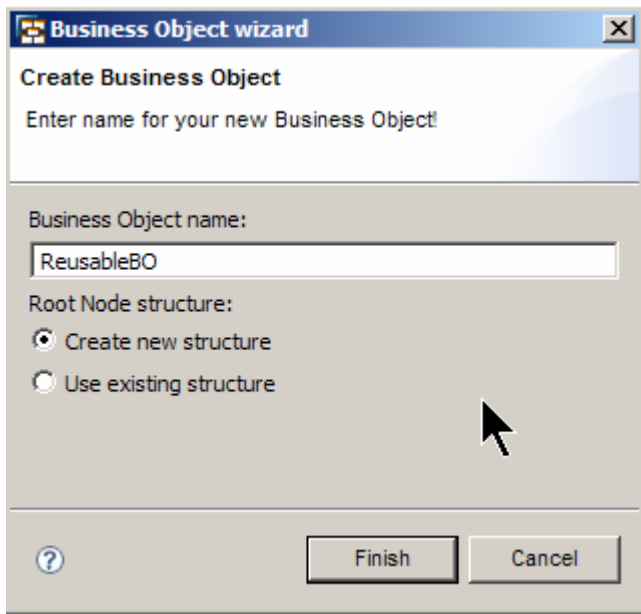
## 3.2 Create the Business Object

Expand the `projecta` node, and navigate to the modeled node

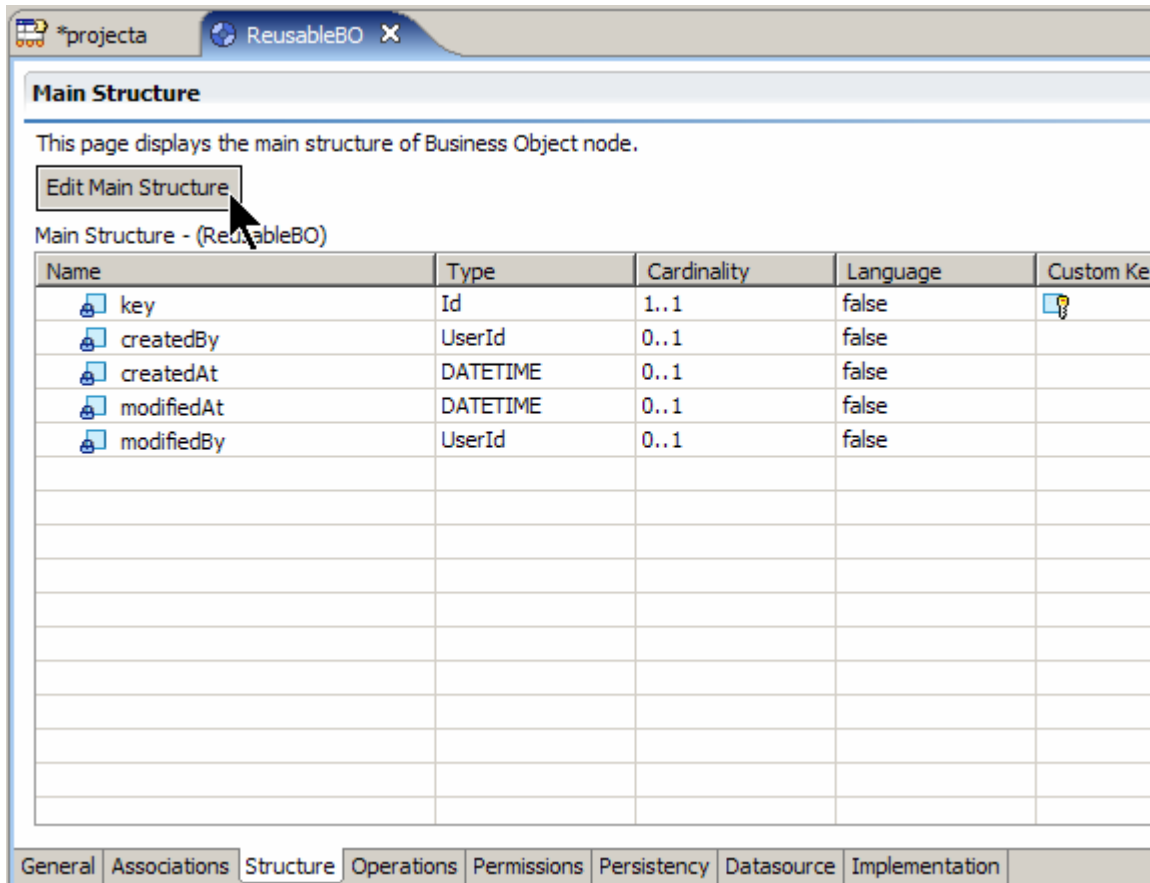
From the context menu choose *New Business Object*.



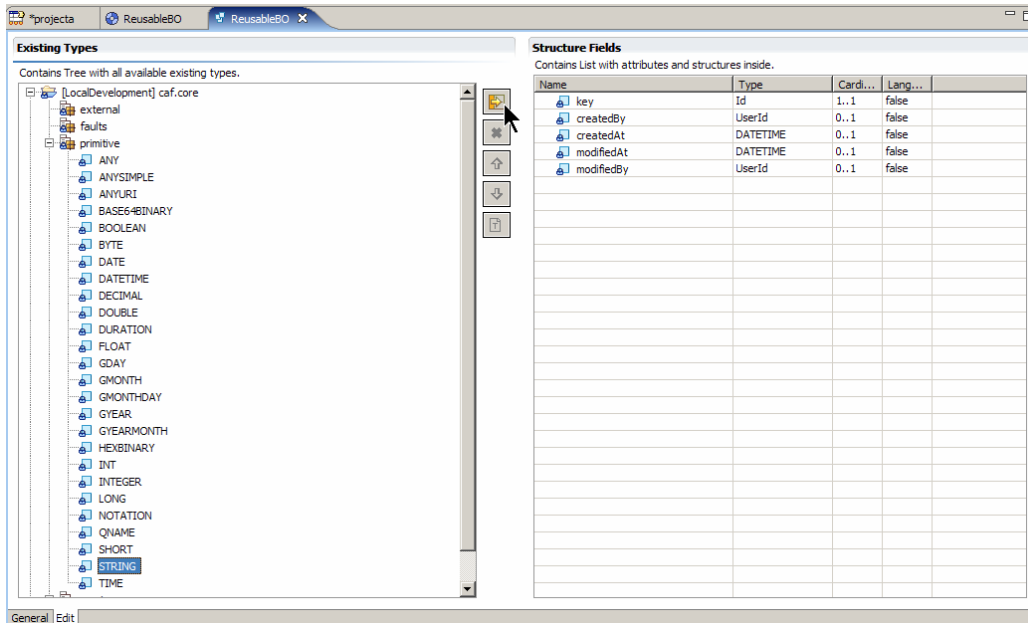
Name the Business Object, select *Create new structure* radio button and click *Finish*.



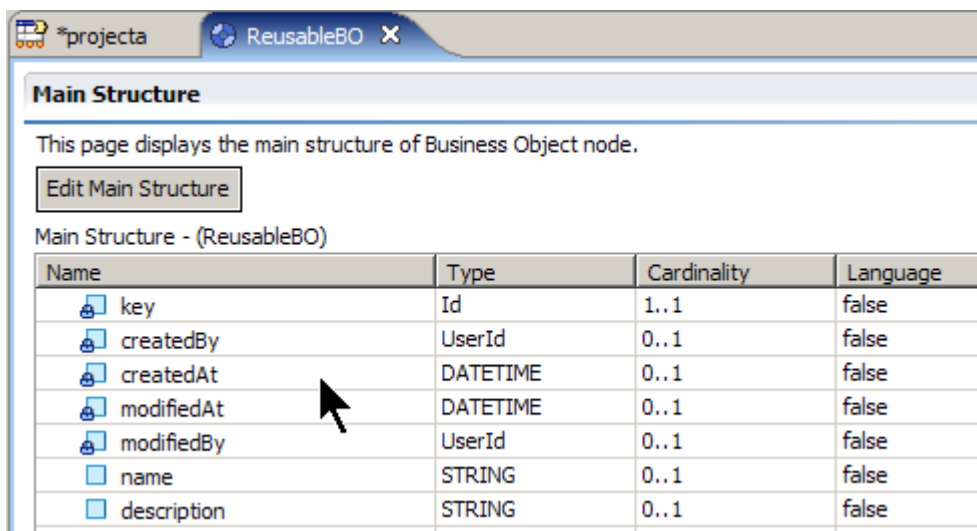
Switch to the *Structure* tab, and click on the *Edit Main Structure* button.



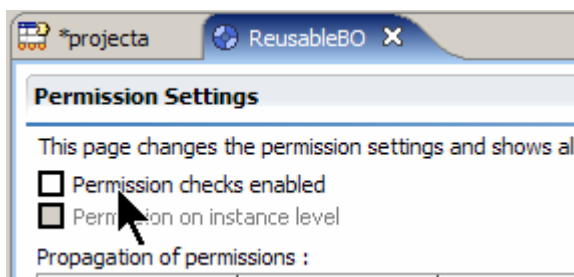
Expand the *caf.core/primitive* tree, and choose the *STRING* data type to add two attributes to the Business Object.



The resultant structure should look like the one shown below.



Switch to the *Permissions* tab, and uncheck the *Permission checks enabled* checkbox.

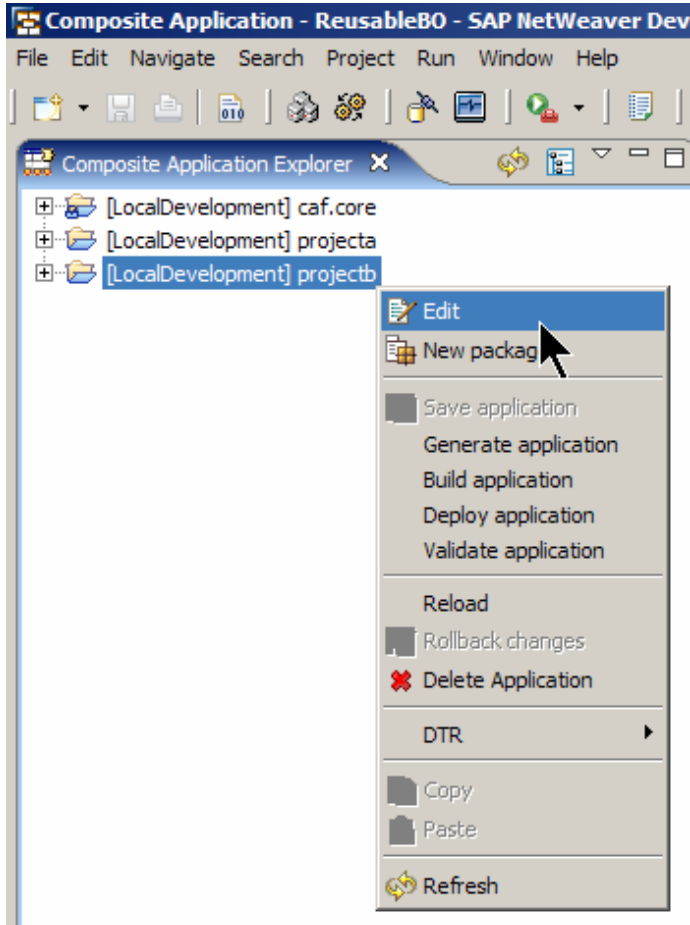


We have now completed the creation of a project with one business object in it. Save the Application, generate and build it.

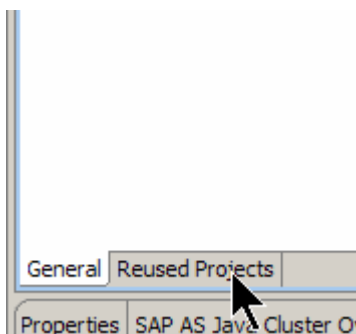
### 3.3 Create the second project (Project B)

Repeat steps in chapter **Error! Reference source not found.** to create another Project. Call this **projectb**.

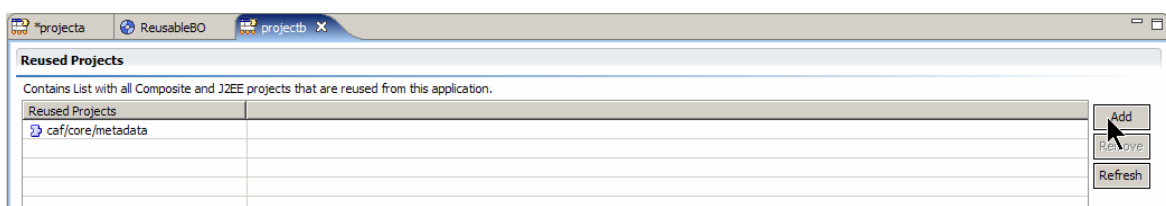
Once done, from the root node open the context menu and choose *Edit*.



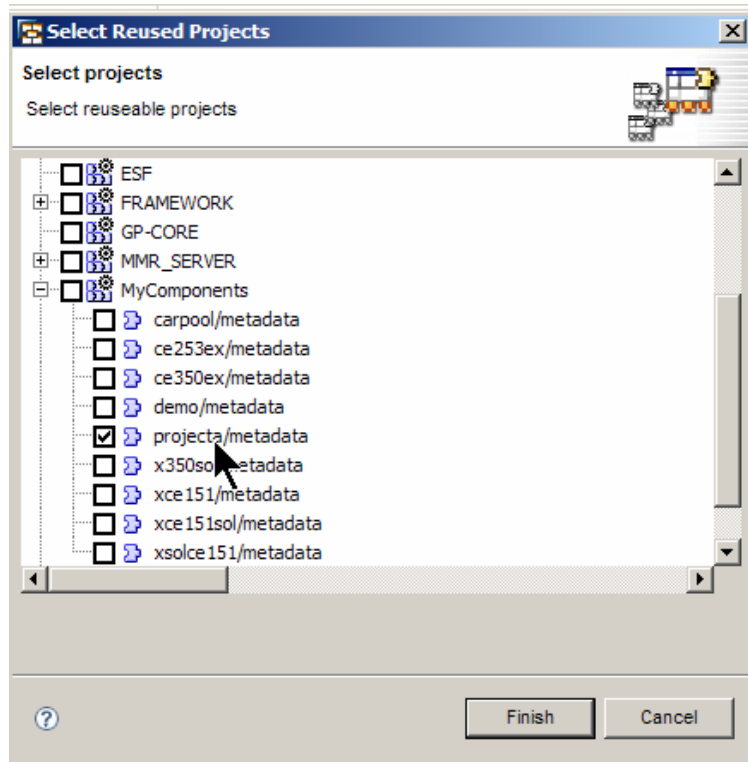
Switch to tab *Reused Projects*.



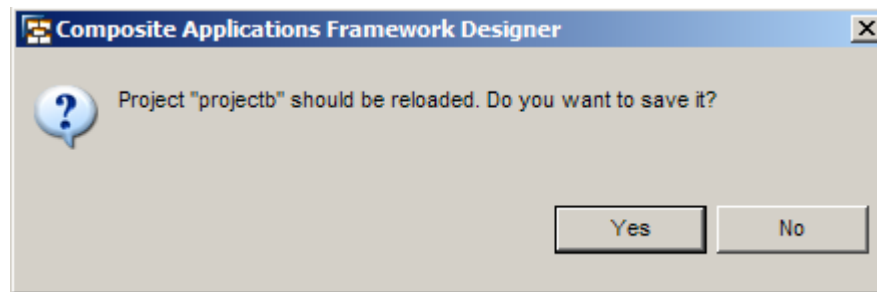
Click on button *Add*.



In the resultant pop up, navigate to the location of the project that contains the business object you want to reuse. Then click on button *Finish*. In this example, we point it to the `projecta/metadata`.

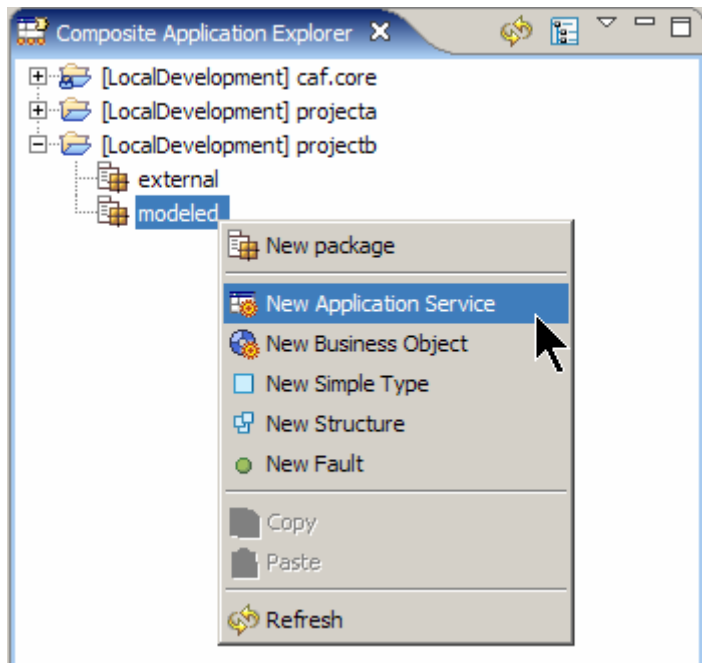


Click on button *Yes* in the message box that pops up.

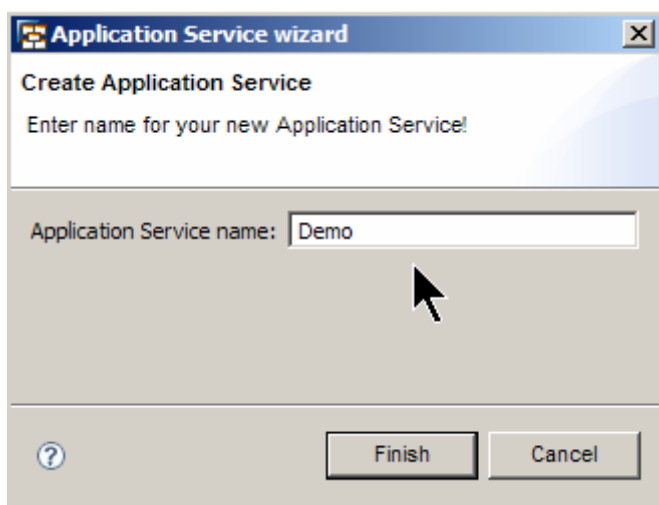


We will now create an Application service in `projectb` which uses the business object from `projecta`.

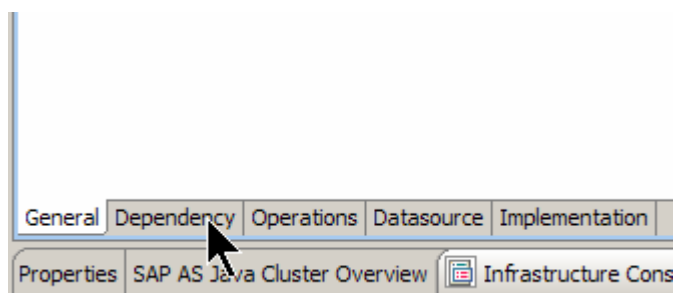
Expand the `projectb` node, and choose *New Application Service* from the modeled node.



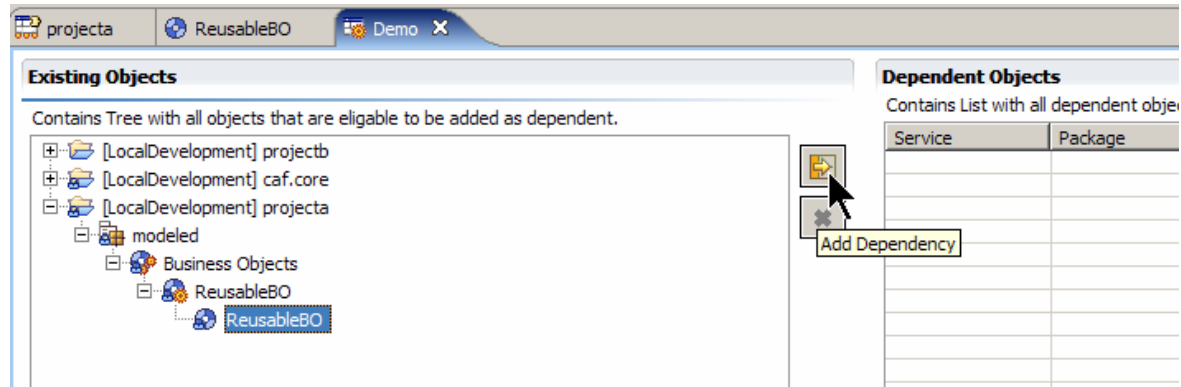
Provide a name for the application service, and click on button *Finish*.



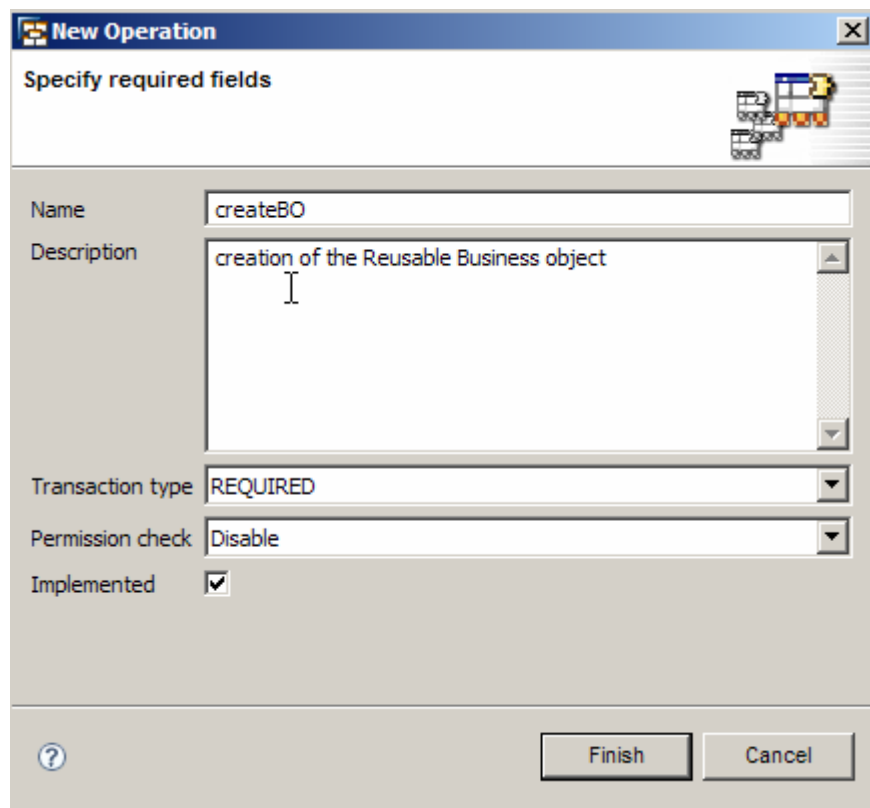
Switch to the *Dependency* tab.



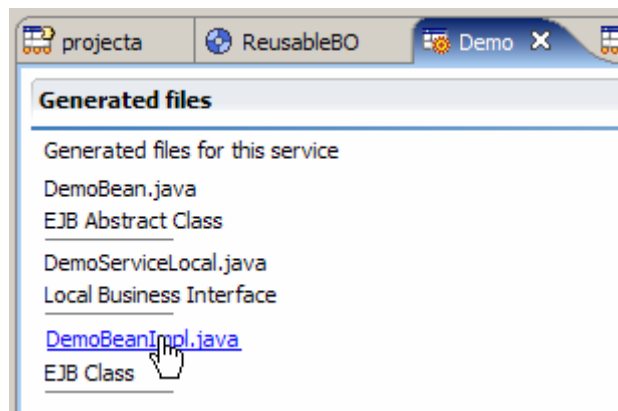
If you expand the tree from **projecta**, notice that now you have access to the **ReusableBO** that we created as part of Chapter **Error! Reference source not found.**. Add this as a dependency.



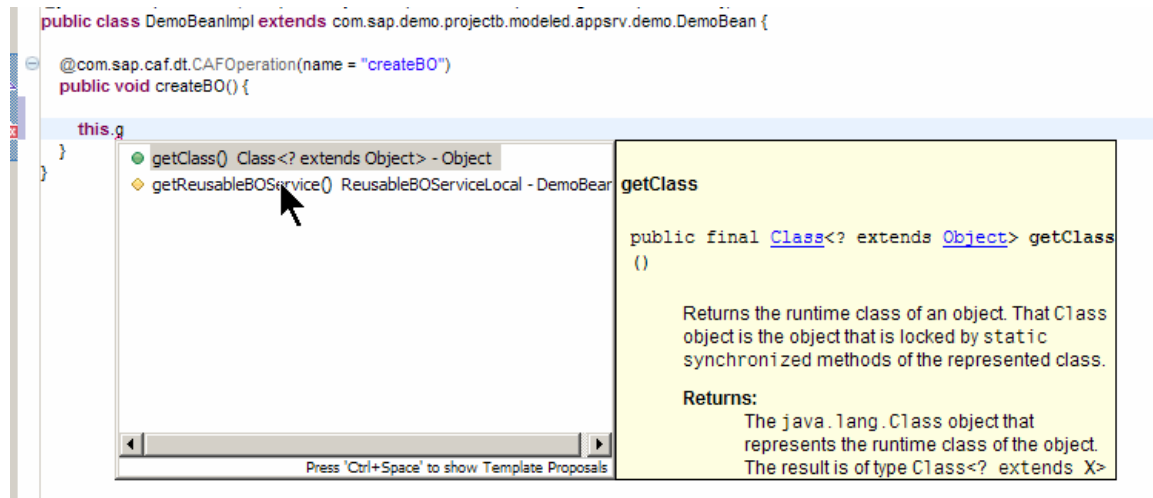
Create an operation by choosing *Add* from the *Operations* tab. Give the operation a name.



Switch to the *Implementation* tab, and select the link *DemoBeanImpl.java*.



You can now get a reference to the Business object from the following method `this.getReusableBOService()`.



```
ReusableBO demo = this.getReusableBOService().create();
```

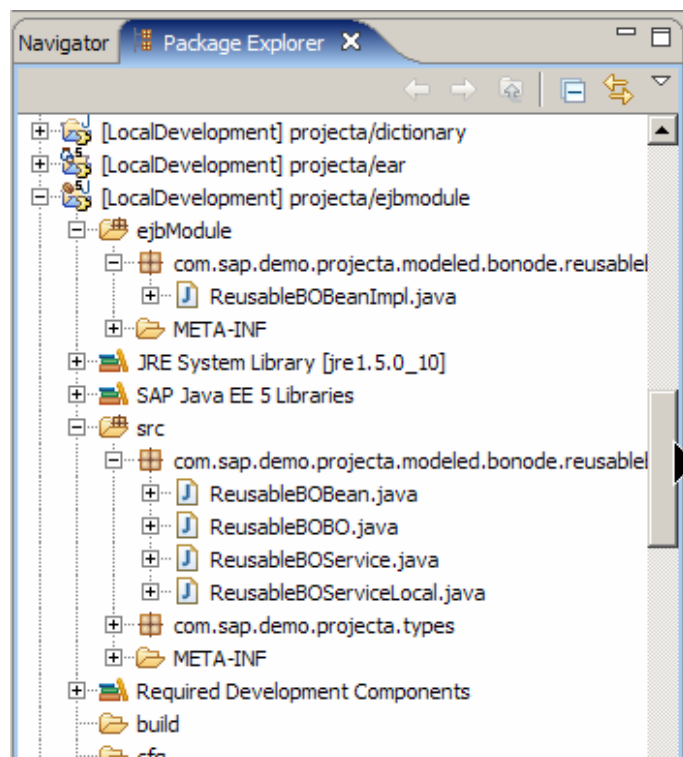
```
demo.setName("abdul");
```

```
demo.setDescription("Demonstration for how to guide");
```

### 3.4 Overriding the Business Object Operations

In certain scenarios it may be required to override the default implementation of the lifecycle methods that the framework creates. In this section we will try to override the create method.

From the package explorer expand the `projecta/ejbmodule`.





Notice that there are two folders where *.java* files are located. One is the *src* folder which is where the framework generates the java files from the metadata. The other location is the *ejbmodule* where the developer can add additional java classes or code.

In our example there is one business object called **ReusableBO** and its corresponding Bean class is **ReusableBOBean.java**. For every Business object, an additional class is created which is called *BOBeanImpl.java* and placed in the *ejbModule* folder.

Double Click on the **ReusableBOBeanImpl.java**. Notice that this class extends the **ReusableBOBean.java**.

```
public class ReusableBOBeanImpl extends
com.sap.demo.projecta.modeled.bonode.reusablebo.reusablebo.ReusableBOBean {
}
```

We just have to override the create method in this class.

```
@Override
public com.sap.demo.projecta.types.ReusableBO create() throws
CAFCreationException {
    //your code to override.
    return super.create();
}
```

[www.sdn.sap.com/irj/sdn/howtoguides](http://www.sdn.sap.com/irj/sdn/howtoguides)