How to Access a Portal Service Within a Web Dynpro Java Application

Based on SAP NetWeaver™ ’04 Stack 09

Jochen Guertler

Copyright

© Copyright 2004 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, and Informix are trademarks or registered trademarks of IBM Corporation 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.

Index and Table of Contents

Index and Table of Contents **3**
Introduction **4**
Defining a Reference to a Portal Service **4**
Gaining Access to the Portal Service **6**
Restrictions **6**

Introduction

In addition to portal applications, it is possible to create “portal services” with functionality that can be reused by different applications. On a semantic level, these portal services are very similar to the standard services of the SAP J2EE Engine; but there are several technical differences as well.

It is possible to get access to such portal services within a Web Dynpro Java application. This document describes how to use a portal service within a Web Dynpro Java application on top of SAP NetWeaver™ ’04 Stack 09.

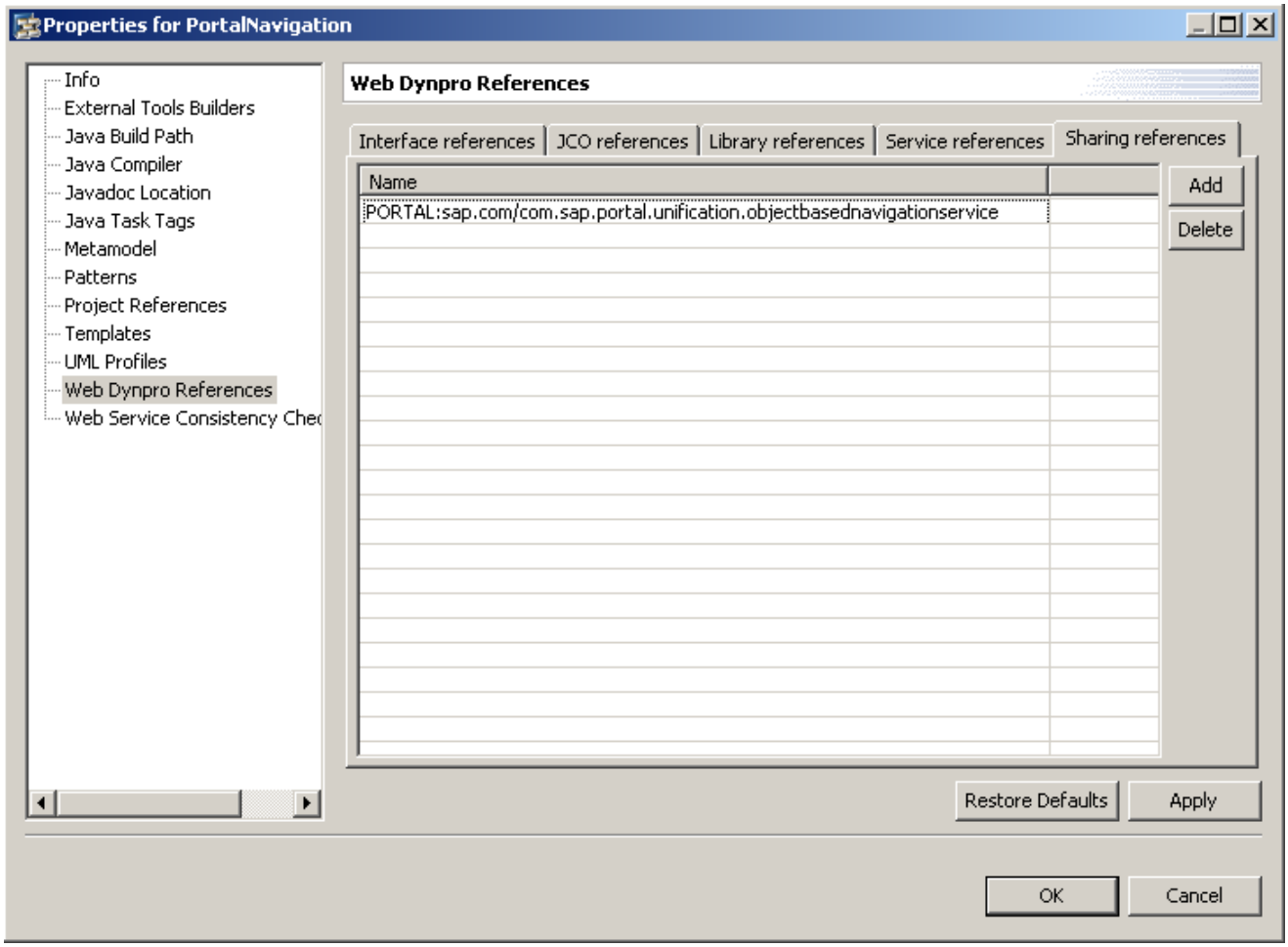
Defining a Reference to a Portal Service

Before you can access a portal service at runtime, the needed reference must be defined. Because a portal service is technically more or less the same thing as a portal application, a sharing reference must be defined for each portal service to be used within a Web Dynpro Java application.

To define a sharing reference, open the “Properties” dialog of your Web Dynpro development component or Web Dynpro Eclipse project (depending on whether you work with the DTR or not).

The sharing references have a certain Web Dynpro reference. The sharing reference must be defined as `PORTAL:<Vendor name>/<Full qualified name of the portal service>.`

The following screenshot shows the example of one sharing reference to a portal service.



After defining the sharing reference, make sure that the necessary class files (those which define the API part of the portal service) are visible in the class path of your Web Dynpro development component or Web Dynpro Eclipse project.

You *must not* store this class files in the “lib” folder of your development component or Eclipse project; doing this could cause classloader problems at runtime.

Gaining Access to the Portal Service

The following code example demonstrates how to get access to a specific portal service:

```
IUserObjectBasedNavigation obnService
      = (IUserObjectBasedNavigation)
      WDPortalUtils.getServiceReference(IUserObjectBasedNavigation.KEY);
```

In this example, the portal service `IUserObjectBasedNavigation` is accessed from a Web Dynpro application.

If there is no more than one Web Dynpro controller using the same portal service, you should store this portal service reference in a certain context node (for example, in the component controller context, or any other custom controller context) All controllers using this service should gain access to it using context mapping to the specified context node.

Restrictions

Theoretically, every portal service could be used within a Web Dynpro application. But there are some restrictions:

- The portal service could only be used if the Web Dynpro application and the portal platform (i.e., the portal itself) run on the same machine.

  In SAP NetWeaver ’04, you have the option whether or not to install the portal platform. The standard SAP NetWeaver Development Workplace (i.e., the standard developer installation), for example, does *not* automatically install the portal platform.

- Even if you use a “full” SAP NetWeaver installation (i.e., the portal platform is installed), keep in mind that you always use the local portal service. If you want to use your Web Dynpro-based portal content within a global or federated portal scenario, there could be problems.

To ensure that you do not use “forbidden” portal services, it is a good idea to contact the developer of the portal service to check for problems (for example, when running in a global portal scenario).