

# SAP NetWeaver Development Infrastructure – Design Time Repository

## Highly Efficient Source Code Management for Distributed Development

### Project Summary

The Design Time Repository (DTR) is a central storage for all kind of source files in various formats. It provides source code management for development based on the SAP NetWeaver platform. The storage is logically kept in a hierarchical files and folders-fashion, however, the physical maintenance is done inside a database which is accessible through an extended WebDAV/DeltaV protocol. The repository serves the SAP NetWeaver Developer Studio through the repository client which offers both programmatic access and a user interface for checking out/checking in source files. Furthermore it provides interfaces for distributing and delivering changes and component releases through the Software Logistic tools provided by the SL-team. The distributed versioning support of DTR with its automatic conflict detection allows to setup arbitrary repository landscapes, for example to integrate bug fixes from older into newer releases in order to reduce the maintenance effort as much as possible.

### DTR Overview

The DTR contains many source files each of them in many versions. However a developer for a specific development task you need only some of these files in a specific version. To achieve this goal the DTR defines logical storage spaces, the **DTR workspaces**. A workspace can be compared to a folder in that it is a list of objects in the database. Each object in the database can in each workspace be referenced in one version only: If the object is updated in the workspace the reference points to the new version. So every developer who accesses the teams workspace to do his or her changes sees the latest version of the sources.

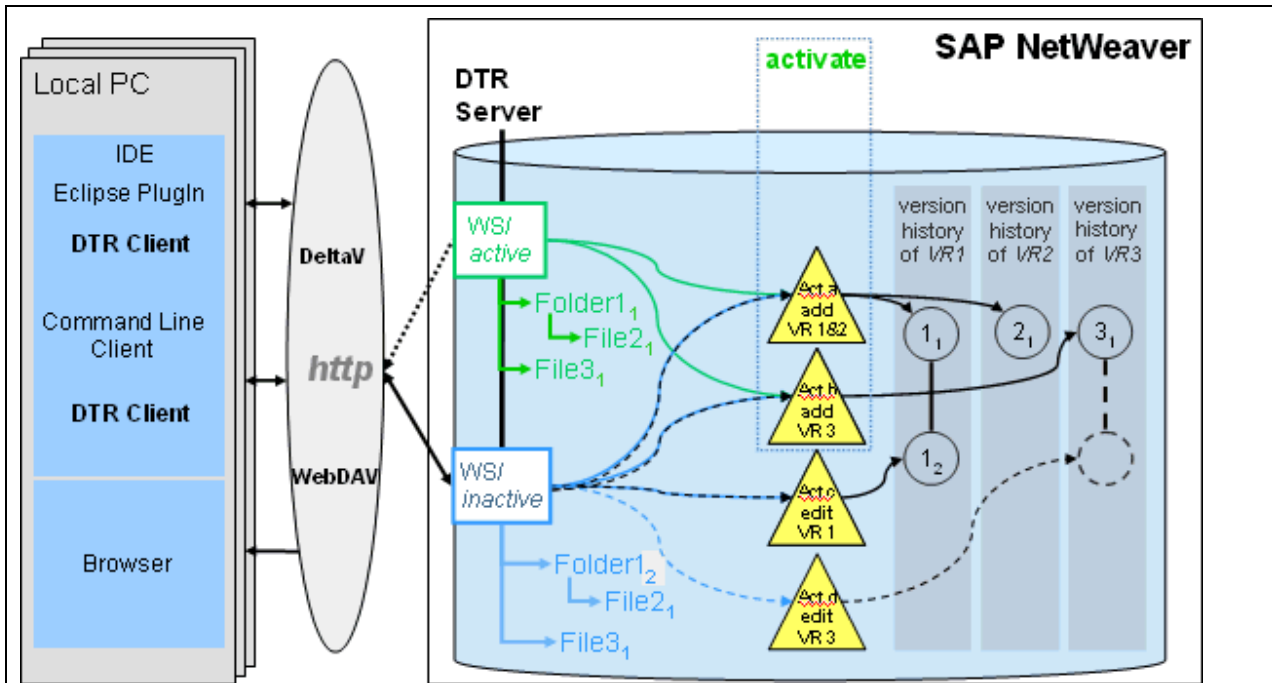
All changes are grouped by **activities**, which contain all new object versions to be stored in the DTR. So all changes to the database content are done in an activity. Therefore a set of activities defines the state of a workspace. Activities are also what is transported when changes are propagated between repositories. When putting them into another workspace, the same mechanisms apply as described above. There are two ways to put versions into workspaces: One DTR can host many workspaces. If two workspaces are in the same repository (as in our example) simply a pointer is set to the same objects in the database, the references objects are physically the same.

There is one problem with changing versions: If more than one developer or party can change the same object a conflict between the two parallel created version occurs (we call this *concurrent development*). A conflict is an uncertainty which of these versions should be used from now on (because both are successors of the same root file, but not of each other). This problem is solved by the DTR by **automatic conflict detection**. In such a case a developer who tries to check-in a conflicting version into the DTR server is asked to decide which version shall be active in the workspace: The version which is already active in the workspace, his or her new version, or a merged version of both. With the grouping of related files in **fusions** the problem of incompatible changes in a complex logical objects such as a Web Dynpro project are addressed.

If objects (source code) are delivered to another DTR, let's say at a customer's site, the objects defined by the state of a workspace are exported and can be imported into the other DTR. There the activities are integrated into a new workspace, containing the content of all active versions of the exported workspace state. So in this new workspace modifications of versions is possible (we call this *distributed development*). In this case, too, conflicts are detected: During this transport all information about the version history of all files are kept so that again with later transports the transported versions can again be placed in the version graph correctly - we call this a **global version history** because it's valid in each instance of the DTR. This

mechanism is used e.g. during the delivery of support packages, so that such a transport doesn't overwrite new versions created by a customer as a modification of a previous delivery.

The following picture describes what you will find in a DTR:



This figure shows the DTR. On the right hand side you see the DTR server, which is accessed through http and WebDAV/DeltaV by the DTR-clients shown at the left. The DTR is a J2EE application running on the SAP NetWeaver using a database. All objects are stored in the database in all versions so each file has a version history. All objects are accessed through DTR-workspaces: A workspace is a logical storage space used by a development team for a specific task such as the development of a software release or a support package. All changes are grouped by activities. A workspace state is therefore defined by the set of activities it contains. Have a look at the example in the picture above:

- There are two workspaces in the DTR, *WS\_active* and *WS\_inactive* both containing files 1, 2, and 3, the index numbers defining the files' versions. The state of a workspace is defined by the activities it contains: *WS\_inact* currently contains activities a, b, and c. This means that *File 1* is used in version 2 (it was edited in activity c), *File 2* in version 1 (added in activity a), and *File 3* in version 1 (added in activity b). There is a new activity 4 containing version 2 of *File 2*, but this is not yet used in the workspace because it has not yet been checked-in (that it's not checked in is indicated by the dotted line; therefore the activity is *open* shown).
- Workspace *WS\_act* contains all files in version 1 because only activities a and b are used.

*DTR architecture: Running on SAP NetWeaver DTR stores versioned resources in a database. Resources are accessed via http-based WebDAV and DeltaV. Activation links the DTR with the CBS.*

What we can learn from that is that each activity may contain one or more file version, and that each activity may be used in one or more workspaces. So, the same object may be referenced in more than one workspace (see Act.a containing file 2 in version 1). As we can see, workspace *WS\_inact* contains two activities a and c, which both contain versions of file 1. This is no problem, because each file version is always put in the correct place in the version graph, so latest version being the version to be used in a workspace - therefore the active version in workspace *WS\_inact* is version 2 of file one.

## Feature List

Design Time Repository provides the following features:

- WebDAV/DeltaV-enabled
- Repository Client (programming API and user interface)
- Index Server (for advanced dependency analysis)
- Propagation interfaces for Software Logistics
- Separation of development states by workspaces
- Sequence-independent propagation of changes
- Global version history allows synchronization of DTR-instances (used e.g. for the modification concept in customer systems)
- Automatic conflict detection
- Combination of sets of files in fusions, which are versioned as a whole

## Related Content

- SAP NetWeaver Development Infrastructure
- Component Model & Component Build Service
- Change Management Service

## Copyright

© Copyright 2007 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.

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.