

**How-to Guide
SAP NetWeaver '04**



How To... Finetune Performance of Portal Platform

Version 0.05 – April 2005

Applicable Releases:

**SAP NetWeaver '04 SAP Enterprise Portal 6.0 on
WebAS 6.40**

© Copyright 2005 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, 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.

1 Introduction

This guide explains how to fine-tune SAP Enterprise Portal 6.0 on WebAS 6.40 to optimize performance. The settings and configuration presented here were tested on a reference hardware system.

In this guide, you will find information on various performance tuning processes about operating systems, SAP Web Application Server (Java), databases, browsers, and suggestions for monitoring the network environment for SAP Enterprise Portal.

The parameters, configurations, and metrics that are presented in this guide can be configured independently, however, consider implementing all the recommendations to enhance performance of the portal.



In different environments and systems, you may have different needs and expectations, and in addition, obtain different results. For this reason, monitor the improvements achieved after making configuration changes.

SAP continuously runs performance tests to guarantee the stability and performance of the portal. Check for up-to-date fine-tuning information using SAP note **772752**. Find SAP notes in SAP service marketplace at: service.sap.com/notes.



Tuning recommendations of hardware devices and mechanisms are not in the scope of this document.

2 Tuning the Operating System

Proper configuration of the operating system is an important factor in the performance of the portal, in most cases however, you may need to adjust the TCP/IP parameters for the portal.

The following sections describe various parameters for different operating systems that support the portal.

See **SAP note 785927** about SAP software on UNIX: OS Dependencies, for updated kernel parameters.

2.1 Tuning Solaris Systems

2.1.1 Tuning TCP/IP Connection Handling on Solaris 8

You may need to tune the way TCP/IP requests are handled, though Solaris generally demands minimal tuning.

For additional information on TCP/IP parameters that apply to Solaris 8, refer to the topic *TCP/IP Tunable Parameters* in the *Solaris Tunable Parameters Reference Manual* at: docs.sun.com/db/doc/816-0607.

When handling a large number of TCP/IP requests, Solaris opens connections for all requests. When the connections are no longer needed by the browsers and Web servers, they are still open.

You must trace the number of connections and modify the settings for them.

To trace and modify the number of TCP connections:

1. At the command line, type the following commands:

```
 netstat
```

```
 ndd -get /dev/tcp tcp_conn_hash_size
```

The parameter `tcp_conn_hash_size` manages the hash table size in the TCP module for all TCP connections. You must reboot the system after changing this parameter.

If the number of TCP connections is high, for example, greater than 5,000, modify the parameter in the file `/etc/system`.

2. Set the table size to 32K entries (default is 512K).

```
- set tcp:tcp_conn_hash_size=32768
```

Note that when the value for the hash table is large, more memory is consumed; so set an appropriate value to avoid wasting memory.

3. Set the *wait interval* to 60 seconds, the minimum value. This value is applicable to Solaris 8 and 9.

```
- set tcp:tcp_time_wait_interval=60000
```

The `tcp_time_wait_interval` is the time in milliseconds a TCP connection stays in TIME-WAIT state. The default value for the time-wait is 4 minutes.

The time-wait value ranges from 1 second to 10 minutes. Too many TCP connections in TIME-WAIT state consume too much memory. You can decrease the value for performance reasons; however, it should not be lower than 60 seconds.

2.2 HP-UX Quality Packs for Java

You must install the latest quality packs for your HP-UX environment before installing your JAVA application. For detailed information about installation of quality packs in HP-UX for Java, see: www.hp.com/products1/unix/java/patches/index.html

2.3 Tuning the Java Virtual Machine

The Java Virtual Machine (JVM) is a program that makes it possible for Java applications to run on a platform. It converts machine instruction into specific instructions executable by a real processor for a specific computer. JVM is based on standards that specify the implementation and deployment of Java applications for a hardware platform.

This section describes JVM parameters for SAP Web Application Server (Java) based on Java development kit (JDK) version 1.4 and above.

2.3.1 JVM Settings



Get updates about JVM settings and their corresponding values using SAP note **723909**, the central note for JVM settings for J2EE 6.30 and 6.40.

In addition, the note contains detailed information about suitable JVM parameters and settings for various operating systems.

3 Tuning the SAP Web Application Server (Java) for the Portal Platform

This section describes key parameters and metrics of the SAP Web Application Server (Java). Finetune the performance of the application server to ensure that the portal runs in a highly optimized environment.

This guide does not describe all the parameters and settings for SAP Web Application Server (Java). Note that SAP Enterprise Portal runs on SAP Web Application Server (Java), which must be configured separately.

The portal cluster environment depends on the proper setup and configuration of SAP Web Application Server (Java). You can configure properties of SAP Web Application Server (Java) in the Config Tool window.

To open the SAP J2EE Config Tool window:

- Run the file *configtool.sh* (*configtool.bat* in Windows) in the folder:
`/usr/sap/<system-id>/jc<instance_number>/j2ee/configtool.`

The settings described in this section are suitable for implementing a portal cluster based on a SAP Web Application Server (Java) cluster.



There are several ways to setup and configure SAP Web Application Server (Java) cluster for the portal. For that reason, the SAP Web Application Server (Java) cluster settings can be configured to suit a customer's needs.

3.1 Using the Template Configuration Tool for a New Portal Installation

Each SAP NetWeaver component has a template configuration tool, which is an application that enables you to check and configure the settings for various hardware dependent parameters for a Java instance, including the following:

- Number of processors (CPUs) to configure for the Java instance
- Average speed in MHz per processor (CPU)
- Amount of physical memory in MB available for the Java instance
- Size of the thread pool
- Java Virtual Machine (JVM) parameters, such as, max heap size in MB per Java dispatcher and server node
- Number of server nodes for the Java instance

The portal template configuration tool is installed with the portal; however, you must load and run it to configure the Java instance and to optimize the performance of the newly installed portal.

For current and up-to-date information about the latest template tool, see SAP note **739788**.

For detailed information about the template tool, go to the SAP help portal at:

help.sap.com/nw04 → *SAP NetWeaver* → *Application Platform (SAP Web Application Server)* → *Java Technology in SAP Web Application Server* → *Administration Manual* → *Server Administration* → *J2EE Engine Configuration* → *Template Configuration Tool*



Use the portal template configuration tool in a newly deployed portal environment.

You should manually configure the SAP Web Application Server (JAVA) to optimize the performance of the portal if there are other NetWeaver components deployed besides the portal.

3.2 Additional Tunable Parameters of SAP Web Application Server (JAVA) for the Portal Platform

You may have to manually fine-tune some of the parameters of the SAP Web Application Server (JAVA) in order to optimize the environment for running SAP Enterprise Portal.

For detailed information about tuning SAP Web Application Server (JAVA), go to the SAP help portal at: help.sap.com/nw04 → *SAP NetWeaver* → *Application Platform (SAP Web Application Server)* → *Java Technology in SAP Web Application Server* → *Administration Manual* → *Server Administration* → *J2EE Engine Configuration* → *Tuning the Engine for J2EE Scenarios*

3.3 Reducing the Network Traffic in a Portal Cluster

3.3.1 Configuring SAP Web Application Server (JAVA) Compression and Caching

You can significantly reduce network traffic when running the portal by fine-tuning both compression and caching.

Compression of HTTP responses affects performance. See SAP note **746666** on how to use third-party Java compression tools in SAP Web Application Server.

SAP Web Application Server (Java) provides compression and caching. To configure compression and caching, use the Config Tool (see Tuning the SAP Web Application Server (Java) for the Portal Platform on page 5) to modify and save changes to the following:

Property	Actions and Notes
<i>HTTP</i>	From the Config Tool window, select: <i><instance></i>  <i>services</i>  <i>http</i> , and change the value of the following properties:
	NeverCompressed To enable script files compression, delete the following content types from the list: *.css, *.pdf, *.js, image, application/pdf, text/JavaScript
	CacheControl 604800 (seconds) Specifies the duration for keeping responses in client cache (if the client is HTTP 1.1 compliant).

Property	Actions and Notes	
	SapCacheControl	604800 (seconds) Specifies the duration for keeping requests in the Internet Communication Manager (ICMan) cache. If similar requests are made within the specified period, the same response is returned directly from the ICMan, without going to the SAP Web Application Server (Java) again. This means that some content remains in the browser's cache for the defined period. After updating portal components, users may experience some problems as the content in the cache has not expired and has not been updated. The system administrator should inform users to manually clear their browser cache, or run a script/tool to do this automatically.
<i>Servlet.jsp</i>	Select: <instance> ↗ services ↗ Servlet.jsp, and change the value of the following property:	
	EnableChunkedResponse	True
	MinimumGZipLength	1024 The default value is 8192. Compressing and decompressing short messages can be slower than sending them as is. Note that any value smaller than 1024 causes the client to consume more resources when decompressing the compressed content.

3.4 SAP Web Application Server (Java) Monitoring Server

3.4.1 Adjusting SAP Web Application Server (Java) Log Severity Levels

Severity levels for log files influence the performance of the following:

- The CPU running the Java instance (central system)
- The J2EE server, due to logging activity

In addition, a large number of log and trace files are generated at runtime. Most of these files are set to log information using the *ALL* severity level, which is not necessary in a production system. Changing the severity log level to *ERROR* significantly improves performance.

See SAP note **763789** for information on changing log levels for SAP Web Application Server (Java).

3.4.2 Disabling the Distributed Statistics Service

In previous versions of the portal (SP9 and below), when you activate the Distributed Statistics Service (DSR), large log and trace files are produced on a node when concurrent users log in and log out at the same time. As the log and trace files grow, they take up valuable hard disk space.

Monitor the size of the log and trace files located in the folders `/usr/sap/ccms/dsr` (UNIX) or `\\<hostname>\prfclog\dsr` (Windows).

You must disable the DSR service to prevent it from increasing the size of the log and trace files.

To disable the DSR service:

1. Run `\usr\sap<system_number>\JC<instance_id>\j2ee\configtool\offlineconfigeditor.bat`

2. Select `Configurations -> cluster_data -> server -> cfg -> services-> PropertySheet dsr-runtime -> <edit-button> -> startup-mode ->`

3. Select `Custom` and enter `manual`, and then select `Apply custom`.

4. Restart the cluster.

For more information, see SAP note **781685**.

4 Tuning the Portal Platform

4.1 Initial Portal Page

The initial portal page refers to the home page or first page that displays in the portal at login.

An initial portal page that has a lot of content, including iViews that start back-end applications, creates a lot of load at login and often results in unsatisfactory performance.

Performance of the portal at login can be influenced by the following:

-  Type of content in the initial page
-  Amount of content on the initial page
-  Type of iViews in the initial page (isolated iViews in IFRAMEs)
-  Number of back-end applications started by iViews on the initial portal page

4.1.1 Some Guidelines for the Initial Portal Page

The following are some guidelines for defining iViews in the initial portal page that can improve portal performance at login.

-  All users should share content in the initial portal page

The initial portal page, or home page, in most cases can be shared by all users. For example, it can be part of the main employee role.

In addition, the content on this page can be virtually the same for all users, although there can be personalization at the page level, such as which iView to show and iViews that show different content to different users.

Type of content in the iViews

The content type for the initial portal page should be informational and not based on applications that change production data. Do not provide iViews with content to back-end systems, or with connections via JCO/JCA and web services for which the responses are delivered on the first page.

When the content type is informational, it can be shared by all users. Such content can come from a cache (that is the content cache for iViews, set by property), which basically requires no computation per user. iViews with information will use the content cache so that the output and effort is shared by many users resulting in a small computation load per user.

Again, as the type of content is information, it does not require much interaction. This results in better performance, better response time, less latency (especially in WANs), less network and CPU load for the same content.

Customized initial page for individual users

iViews in the initial portal page should not be dependent on each user's needs, but only on personalization parameters. For instance, each user can personalize the city in a weather iView, but the weather is the same for the same city.

The initial portal page should contain information that is mostly the same for all users, and the iViews should be embedded (not isolated) on the page.

Failure to follow these guidelines can result in inadequate performance at login.

4.1.2 JCO iViews in the Initial Portal Page

JCO iViews access SAP systems via a JCO client, ***com.sap.portal.runtime.application.jcoclient***.

Including JCO iViews in the initial page is not recommended; however, if you have to include such iViews, then make use of the following:

Connection pooling

A means of saving time in establishing connectivity with back-end systems by collecting connections that have been used earlier, for later use. For example, iViews that are executed in sequence can benefit from the pool, and respond faster resulting in enhanced performance.

Caching levels

Controls caching for a portal object. The value for this property can be either shared, or group.

Implementation of both the connection pooling and caching levels is done with API methods to interfaces with JCO iViews.

4.2 Optimizing Portal Catalog Administration

For administration tasks in the Portal Catalog, it is recommended that you organize the content in a folder taxonomy.

Placing many objects in a single folder increases the time it takes for that folder to load. The recommendation is to put no more than 50 objects in a single folder.

4.3 PRT Configuration

The file *prtDefault.properties* contains various properties that can be configured for the PRT. You can find the file in the folder:

`\usr\sap\`

The following table shows some of the properties in the file *prtDefault.properties*:

Setting	Action and Notes
<i>Async.Response.Pools.size</i>	<p>The number of buffers that the Page Builder can use to process asynchronous rendering of pages.</p> <p>async.response.pool.size=100</p> <p>Each buffer is used during the rendering of a portal page. Increasing this value increases the number of pages to be rendered asynchronously.</p> <p>A large value can have significant effect on the system, as it is affected by available system resources, such as memory.</p>
<i>PRT monitor</i>	<p>Enables the PRT monitor, which helps identify bottlenecks in performance.</p> <p>In productive environments, we recommend the following:</p> <p>monitor.off=true</p> <p>To enable the monitor:</p> <ul style="list-style-type: none"> • Change the setting to <i>false</i>. <p>To view the monitor output:</p> <ul style="list-style-type: none"> • Log on to the portal as an administrator, and navigate to: System Administration ↗ Monitoring ↗ Portal ↗ Components Overview System Administration ↗ Monitoring ↗ Portal ↗ Threads Overview

4.4 PRT Cache

When the portal has been configured to use the database for caching data, it stores the content of an

iView in the database.

The size of the data stored depends on the size of the content in the iView. A large content in the cache affects the performance of the portal.

This occurs especially when using an Oracle database for the portal in a UNIX system.

For more information, see SAP note **792154**.

4.5 Optimizing the Client Framework Script for Portal Pages (JavaScript and CSS)

The Client Framework includes JavaScript installed out-of-the-box with SAP Enterprise Portal. In a production environment, you can change the configuration settings for the JavaScript that is delivered from the server to clients.

Change the configuration settings for the JavaScript using the *scriptset* property. This property defines the type of display for the JavaScript, so that based on the defined level for the Client Framework, the suitable JavaScript Include is automatically integrated in each portal page.

The default configuration sets the Client Framework to contain comments and formats that are easy to read. Changing the default to **Optimize** eliminates comments and unnecessary spaces that will be displayed.

For additional information on how to configure the Client Framework, go to the SAP help portal at help.sap.com > *SAP NetWeaver* > *People Integration* > *Portal* > *Administration Guide* > *Content Administration* > *iViews* > *Client Framework for Administrators* > *Configuration*

5 Browser

5.1 Setting the Browser Compression

New browsers support the HTTP 1.1 standard known as content-encoding. Essentially, the browser indicates to the server that it can accept content-encoding, and if the server is capable, it will then compress the data and transmit it. The browser decompresses the data from the server, and then renders the page.

Only HTTP 1.1 compliant clients, request compressed files. Clients that are not HTTP 1.1 compliant, request and receive the files uncompressed, and do not benefit from improved download time from HTTP 1.1.

Internet Explorer versions 4 and above, Windows Explorer, and My Computer are all HTTP 1.1 compliant clients by default.

When using Microsoft® Internet Explorer as your client browser, activate HTTP 1.1.

To set the compression filter:

1. From the browser menu, select *Tools* and then *Internet Options*.
2. Choose the *Advanced* tab and select *HTTP 1.1*. Make sure the *Use HTTP 1.1* checkbox is selected.
3. If your network operates with a proxy server, verify that *Use HTTP 1.1 through proxy connections* is also selected.
4. Click *OK*.

Browser caching options

MS Internet Explorer 5.0, 5.5, 6.0

Browser Caching

You can improve the browser response time by using the browser cache. This enables your browser to cache resource files, including the following file types: images, CSS, JS and text.

For detailed information on browser caching, go to the SAP help portal at: <http://help.sap.com/nw04> > *SAP NetWeaver* > *People Integration* > *Portal* > *End User Guide* > *Accessing the Enterprise Portal* > *Setting Internet Browser Options*.

To use browser cache in Internet Explorer:

1. From the *Tools* menu > *Internet Options* > *General* tab, click *Settings*.
2. Under *Check for newer versions of stored pages* in the *Settings* dialog box, select either *Automatically* or *Every Time you start Internet Explorer*.

Do not choose *Every visit to page*.

3. Click *OK* twice to save the settings for the browser cache.

6 Tuning the Internet Transaction Server (ITS)

The Internet Transaction Server (ITS) is a gateway mechanism located between one or more Web servers, and one or more SAP application servers.

By default, HTML page compression is enabled between ITS and the browser. The compression technique is based on a compression algorithm that is also part of major browsers.

For up-to-date information about ITS, see SAP note **419297**.



Only the data that is processed by the AGate is compressed.

The dynamic data that is processed through the WGATEURL (default/scripts/wgate) and static data that is called in the MIMEURL (default /sap/its/mimes) are not compressed. You need other hardware or software tools for compressing these mime types.

Third-Party Compression Tools

There are compression tools for Web servers, such as Apache, IPlanet or Microsoft IIS.

With IIS 5.0, Microsoft delivers an additional filter, *compression.dll*. By default, this ISAPI filter is disabled. For more information about using this ISAPI filter compression for *.js*, and *.css*, see:

www.microsoft.com/technet/prodtechnol/iis/maintain/featusability/httpcomp.asp



Problems can arise if the compression tool compresses the data sent by the AGate a second time. In this case, either turn off the compression for the WGATEURL or set `~http_use_compression` to 0.

7 Network Tuning

7.1 Network Analysis Tools

7.1.1 Analysis Tools

The analysis tools that come with the SAP Enterprise Portal are briefly described below. Several other non-portal-specific tools are useful for performance analysis of the portal.

Network Sniffers

Trace all kinds of network communication directly on the network card: HTTP, LDAP, database, CIFS/SMB (remote shares). By default, the tool displays just network packets and interprets them according to the used protocol. One common feature is to recombine the packets to the TCP stream again, such that, for example, the HTTP content can be displayed. Due to packet switching, the network sniffer must be active on one of the two communication endpoints of the network communication to be analyzed. With network switches, other hosts will normally not see any IP packets that are not addressed to them.

Example Tools:

- **Ethereal** (www.ethereal.com) – Open source, underlying `wireshark` runs only on single-CPU hosts (that is, can normally not be used on the portal server itself).
Usage: capture data (maybe directly with filter), find relevant packages, follow TCP stream to display the complete conversation.
- **HttpWatch 3.2** (www.simtec.ltd.uk/buy) – HttpWatch integrates with Internet Explorer to provide HTTP monitoring without the need for separately configured proxies or network sniffers. Interact with a web site and the tool displays a log of requests and responses alongside the web page, and shows interactions between Internet Explorer and the browser cache. All data can be saved to XSL file for further processing.
- **HttpLook 1.3** (www.httpsniffer.com/) – HTTPLook gives you a way to test and debug internet applications, working over HTTP protocol, such as web sites, web services and their clients, allowing you to trace messages by which these applications communicate. HTTPLook captures, without any modification, data traffic of all internet applications, including encrypted SSL traffic of Internet Explorer. It parses HTTP messages, separates headers and entities, decodes compressed entities, and saves entities in your file system. You can sort, filter and search messages.

7.1.2 Optimization Tools

- **JSMIN** (www.crockford.com/javascript/jsmin.html) – JSMIN is a filter that removes comments and unnecessary whitespaces from `JavaScript` files. It typically reduces file size by half, resulting in faster downloads. It also encourages a more expressive programming style because it eliminates the download cost of clean, literate self-documentation.
- **CSS Optimizer** (lumpcakes.co.uk/css/optimiser) – This tool takes a CSS file, parses it and outputs a leaner version, removing any redundancies and attributes that are not needed (removes comments and white space, converts multiple background, font, margin, padding and list attributes into a single attribute).

www.sdn.sap.com/irj/sdn/howtoguides