

Technical Infrastructure of SAP Enterprise Portal 6.0 on SAP Web AS 6.40



**SAP Enterprise Portal 6.0
on SAP Web AS 6.40**



Document version 2.0

Copyright

© 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, 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.

Icons in Body Text

Icon	Meaning
	Caution
	Example
	Note
	Recommendation
	Syntax

Additional icons are used in SAP Library documentation to help you identify different types of information at a glance. For more information, see *Help on Help → General Information Classes and Information Classes for Business Information Warehouse* on the first page of any version of *SAP Library*.

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.
Example text	Emphasized words or phrases in body text, graphic titles, and table titles.
EXAMPLE TEXT	Technical names of system objects. These include report names, program names, transaction codes, table names, and key concepts of a programming language when they are surrounded by body text, for example, SELECT and INCLUDE.
Example text	Output on the screen. This includes file and directory names and their paths, messages, names of variables and parameters, source text, and names of installation, upgrade and database tools.
Example text	Exact user entry. 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.

Contents

SAP Enterprise Portal Technical Infrastructure.....	5
Portal Architectural Overview.....	5
Portal Platform.....	5
Planning the Portal Landscape.....	7
Distributing the Components of SAP Enterprise Portal.....	10
Planning the Portal Cluster.....	11
Planning the Cluster Configuration	13
Preparing the Portal Cluster.....	13
Clustering the Portal Platform.....	13
Identifying and Removing Single Points of Failure (SPOF)	14
Load-Balancing the Portal with the SAP Web Application Server	15
Managing Resources Through the SAP Web Application Server	15
Performance and Availability in the Cluster.....	15
Knowledge Management and Collaboration in the Portal Landscape	16
Content Management and Collaboration	16
Content Management and Collaboration Distribution	17
Content Management and Collaboration in a Cluster	19
TREX.....	22
Java Client	22
Web Server with TREX Enhancement	22
Queue Server	22
Preprocessor	23
Index Server	23
Name server	23
TREX Distribution.....	23
Network Landscapes.....	25
Portal in the DMZ.....	26
System and Portal Administration	27
Security	27

Document History

Document version	Description
2.0	<ul style="list-style-type: none"> ▪ Added reference to detailed information about the template configuration tool. ▪ Updated references to headings in the Help Portal.
1.50	Previous official release of this document.



SAP Enterprise Portal Technical Infrastructure

Purpose

This guide provides guidelines for planning your technical landscape for the portal. It shows a range of solutions, including clustering and load-balancing methods and mechanisms.

In addition, this guide explains approaches to scaling, and optimizing the performance of the portal.

This guide also describes deployment and implementation of the portal in a secure network landscape.

Implementation Considerations

Partners, consultants, and implementers can use this guide to provide a portal that is secure, and highly available with optimized performance.

This guide should be used as a supplement to the portal installation guides. For information go to: service.sap.com/instguidesNW04 → *Installation* → *SAP EP* → *Installation Guide - SAP EP 6.0 (SR1) on WebAS 6.40*



Portal Architectural Overview

The portal platform is one of the building blocks of SAP NetWeaver into which other NetWeaver components can be integrated, such as:

- Knowledge Management platform
- Collaboration

Portal Platform

The portal platform is deployed on the SAP Web Application Server (JAVA), which implements the J2EE engine based on J2EE standards.

The platform consists of the following:

- **Portal Framework**

The portal framework is a virtual environment that enables the portal both as a development and a runtime environment.

Applications that run in the portal, such as iViews, are implemented by portal components and portal services including the Portal Content Directory (PCD), which interfaces with a database system.

Portal services, such as unification, enable the implementation of the interfaces that allow Drag&Relate processes among iViews and business objects for back-end systems, and enables connectivity to Enterprise Information Systems for connectors.

In addition, it supplies the Connector Framework, which is comprised of adapters that provide connectivity between an Enterprise Information System (EIS) and the portal.

The following components are required in a landscape of the portal:

- **Database:** The portal shares the same database and schema for SAP Web Application Server. The database stores various data and metadata required by the portal at runtime.
- **User Persistence Store:** A storage area containing information about users, such as directory servers. This system works with the mechanism that implements security systems for authentication and authorization: the User Management Engine (UME), an SAP proprietary mechanism that allows access to various content, using single sign-on (SSO) for users.
- **Java Development Kit (JDK):** The Java runtime environment required by the Java software and components in the portal.

Identifying Portal Resources at Runtime

Categorizing the resources required at runtime lets you identify those resources that are common to the platforms of SAP Enterprise Portal and can be shared, distinguishing them from those that should be distributed over several physical machines.

Portal resources are of two types: runtime resources and the storage resources used during runtime.

Runtime Resources

Runtime resources consist of portal components and services that generate the portal framework, assemble portal pages and iViews, and those that manage themes and styles. These resources use stored data in the database system and in the user persistence store to implement the user interface of the portal.

The portal runtime resources are:

- The J2EE engine of the SAP Web Application Server
Every portal deployment requires an instance of the J2EE engine.
The portal infrastructure is scalable, and a highly available portal can be implemented using the load-balancing and clustering mechanisms of the J2EE engine of the SAP Web Application Server.
- Portal Framework
The portal framework provides the working environment for the portal. It allows a user to work as a portal administrator, a developer, or an end user, depending on the user's logon profile.
Inter-operation and integration between the components and services that implement the portal infrastructure and functionality exist in this environment.

Runtime Storage Resources

The storage resources maintain stored data that is used at runtime. These resources provide shared functionality to the platforms of SAP Enterprise Portal.

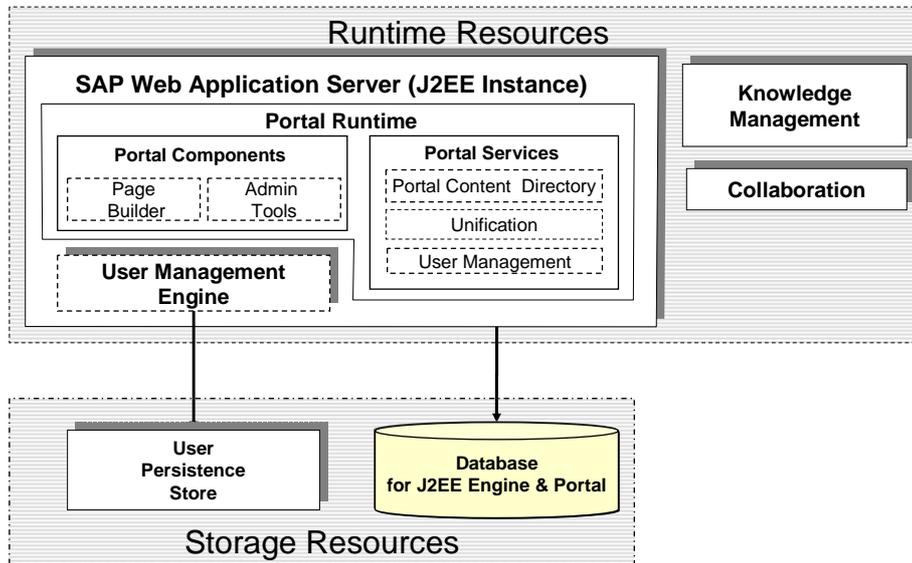
The storage resources include the following:

- Database (RDBMS)
This is the database for SAP Web Application Server (JAVA), which is usually located in a separate machine.
The portal stores various data and metadata for the Portal Content Directory (PCD) in this database.
In addition, the database stores portal runtime objects, including role definitions, page-to-role relationships, deployable portal archive (PAR) files, templates, and personalization data.

- User Persistence Store

Refers to users-related data stored in one or more repositories. A user-related data repository might be a database, a Lightweight Directory Access Protocol (LDAP) directory service, or an SAP R/3 system.

The following figure illustrates the runtime resources and the runtime storage resources of the portal:



Planning the Portal Landscape

You can deploy the portal in stages, for each of which you will need an implementation plan that is distinct to the landscape.

The following are the key stages in which you plan and implement the portal landscape:

1. Development

The development stage consists of installation of the portal platform for content development.

2. Integration and Testing

Integration and testing stage involves fine-tuning and troubleshooting issues, such as validating installations and testing the content developed during the development stage.

3. Production

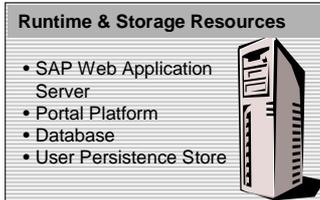
In the production phase, you scale the portal and implement high availability solutions.

As you deploy the portal in stages, any change implemented after completing the production stage, such as applying upgrades, support packages, patches, and other applications changes the environment. It is recommended that, you first test the changed landscape before deploying the change in the production environment.

Development Stage

In the development stage, your portal landscape can consist of one or two servers. This usually depends on the number of users and the amount of content.

Technical Landscape: Deploying the portal for content development in a single server



Although the development landscape can consist of a single physical server, it is recommended to deploy the portal in two physical servers: one server to run the portal and the JAVA instance, and the other to maintain the storage resources required at runtime, such as the database.

You may want to prepare this landscape for a productive environment by vertically scaling the portal. You do so by adding more server processes (J2EE server nodes) to the JAVA instance in a single physical server.

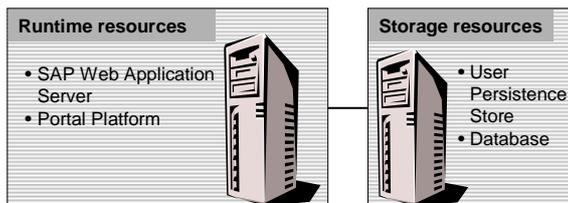


You can configure the JAVA instance to optimize performance of the newly deployed portal landscape using the template configuration tool.

For more information about the template configuration tool go to the SAP help portal at: help.sap.com/nw04 → *English* → *SAP Library* → *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*

The following is an illustration of the components installed for a development landscape:

Technical Landscape: Deploying the portal for content development



At the development stage, the primary focus is to create content (iViews) for various users. Two kinds of portal users will be working in this landscape: portal administrators, and content developers.

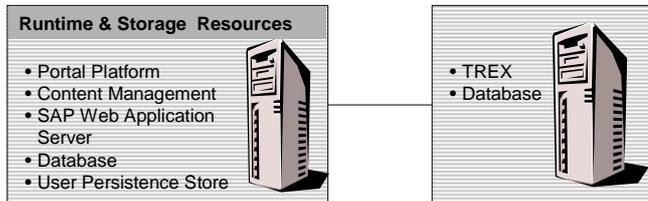
Integration and Testing

Integration and testing is a continuous process that must take place in any landscape. It follows the development stage, such as changes that are introduced after the initial portal deployment.

For instance, when you install Content Management in addition to the portal platform, you need to test both components before expanding the portal environment for production.

The illustration below is an example of an integrated portal landscape:

Technical Landscape: Deploying the portal for integration and testing



Portal administrators play an active role in this stage, such as assigning iViews to users, and ensuring that users have the proper security credentials to content in the iViews that are assigned to them.

Production Stage

The production stage is an extension of the development stage. It begins when content development, and integration and testing have been successfully completed.

The landscape implementation in a production environment usually consists of two or more physical servers. The database and the user persistence store are placed separately from the portal platform installation.

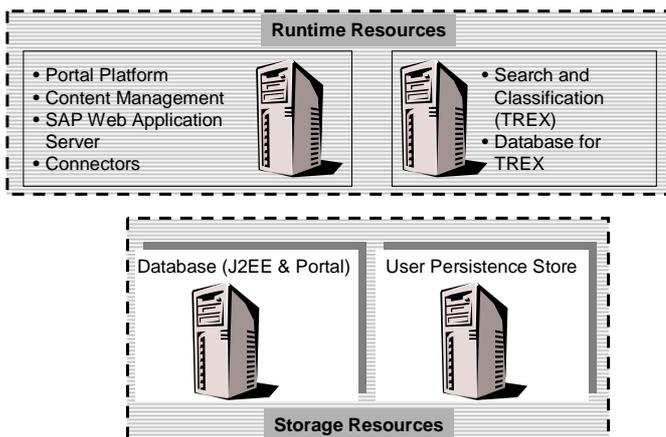
In this stage, portal implementation can include several deployment cycles, where each cycle focuses on the deployment and implementation of specific SAP Enterprise Portal components, such as Content Management, Collaboration, Business Packages and Connectors.

TREX, which is both a processor and network-intensive application (depending on whether or not it is indexing remote shares), must have a separate server.

Planning the production environment involves several experts including implementers, system administrators, database experts, and network security experts.

The following figure is a landscape in the production environment showing runtime resources and storage resources:

Landscape: Deploying the portal in a production environment



At the production stage, consider the following:

- First scale the portal vertically, by increasing the number of server processes within a single machine, and if necessary add more CPU and memory.

Later, scale the landscape horizontally by adding more machines (each additional machine is scaled vertically, as described above, to optimize the JAVA instances). For more information, see Scalability and Availability Considerations, on page 12.

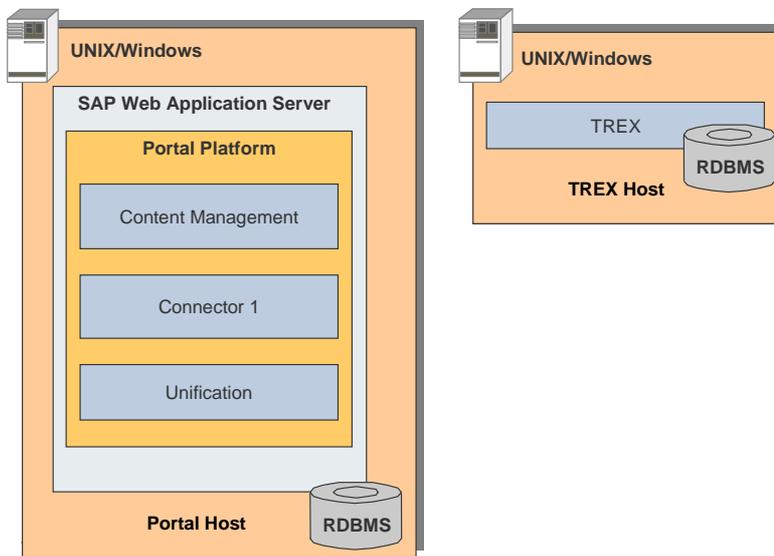
- To make the reliable portal landscape highly available, remove the single points of failure (SPOF) that you identify in the landscape. For more information, see [Identifying and Removing Single Points of Failure \(SPOF\)](#) on page 14
- To increase security, locate the production system behind a firewall. For more information, see [Network Landscapes](#) on page 25.
- Fine-tune the scaled and highly available portal landscape to optimize performance. See SAP note number **772752**.



Distributing the Components of SAP Enterprise Portal

The infrastructure of the portal allows you to deploy the portal resources separately.

The following figure illustrates deployed components of the portal for development and testing:

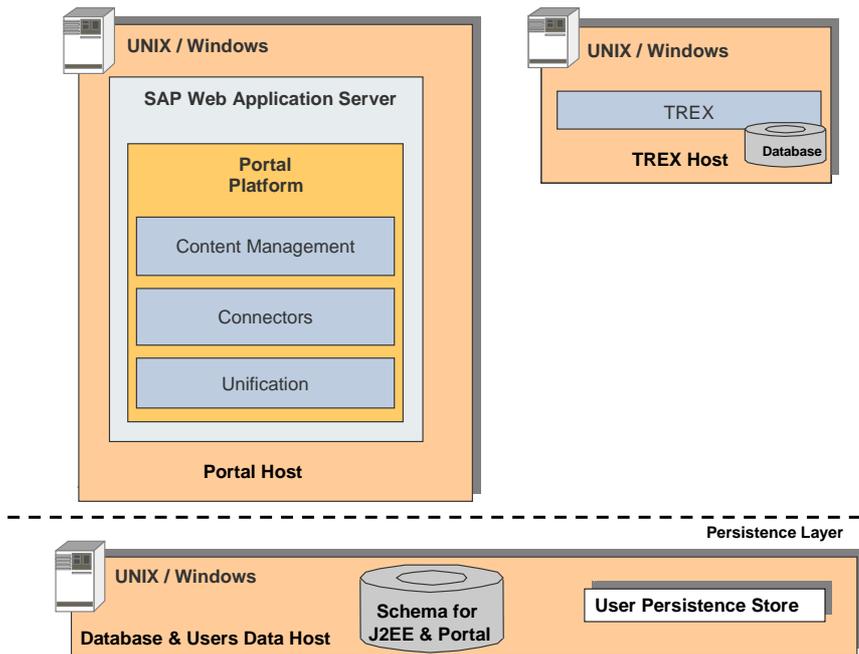


The portal host in the figure above includes portal resources, such as the database (RDBMS), and the user persistence store (user-related data) on a single machine. Content Management is always deployed on the portal host, however, it is recommended to deploy Search and Classification (TREX) on a separate machine.

Setting up the Persistence Layer in a Separate Machine

An alternative method of deploying the portal for development and testing is to locate the database, and the user persistence store on a machine other than that of the portal host.

The following illustrates the deployment method of SAP Enterprise Portal:



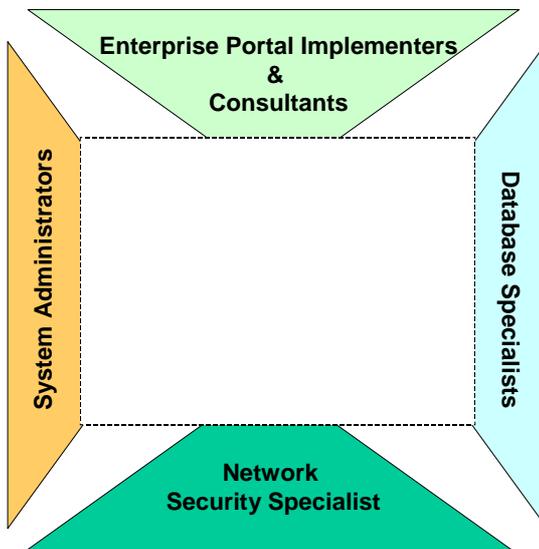
When a single portal host has to serve many portal clients, as in the above landscape, demand on the resources and processes can become an issue.

You can add more servers to host additional portals, forming a cluster of portals. Implementing hardware, or software load-balancing solution for the portal cluster makes it highly available and accessible to a large number of users, or portal clients.

Planning the Portal Cluster

When you plan to deploy the portal on several machines to form a cluster, you need to consider the possible load on the network infrastructure.

Planning a cluster for the portal is a joint effort among several specialists, such as security experts, systems administrators, database specialists, Enterprise Portal implementers and consultants. The following figure illustrates the set of specialist who are likely to collaborate in planning the cluster:



Planning can be in stages:

1. SAP Enterprise Portal implementers, consultants, security experts, and database specialists determine the customer-specific network requirements.
2. SAP Enterprise Portal implementers, system administrators, security experts and database specialists work to meet the defined requirements, that may consist of the following:
 - Network topology and network components (routers, hubs, switches, servers, if necessary with multiple network connections)
 - Network configuration of the servers (IP addresses, routes, host names)
 - Installation and configuration of SAP Web Application Server, Portal Platform, Content Management, Collaboration and TREX
 - Testing the connections and conditions under full network load

When planning the portal cluster environment, consider installing the database, and the user persistence store (storage for user-related data) separately, each on a dedicated server.

Scalability and Availability Considerations

You can scale the portal by increasing the JAVA instances in either a single machine, or over several machines. Either way, you can implement a load-balancing solution that enables the portal to handle a large number of users.

You can implement the load-balancing features of SAP Web Application server for the cluster of portals. Alternatively, you can implement an external load-balancing solution of third-party hardware or software vendors.

However, note that some load-balancing solutions have limitations on the number of physical servers in the cluster environment.

Front-End Devices

Issues relating to front-end devices for SAP Enterprise Portal include the following:

- Stable TCP/IP connection to front-end devices, such as browsers
- Fine-tuning of front-end devices, such as setting the browser compression to enhance performance, and the effects of upgraded installations.

For more information about fine tuning the portal landscape, see SAP note number **772752**.

Backup and Restore Strategies

In a production environment, consider a backup and restore strategy. Strategies may include both offline and online backups.

If the data is backed up from a central point over the network, there is a heavy network load. If the backup occurs during operational hours of the portal (which includes background processing), consider the effect of increases in the delay time and limitations on the bandwidth that can affect the overall performance.

For more information on *Backup and Restore strategies*, refer to the section on *Backup and Recovery* on SAP help portal at: **help.sap.com/nw04** → SAP Library → *SAP NetWeaver* → *Application Platform (SAP Web Application Server)* → *Java Technology in SAP Web Application Server* → *Administration Manual* → *Server Administration* → *Backup and Recovery of the SAP Web Application Server (Java)*

Additional information on *Backup and Restore strategies* exists on SAP help portal at: **help.sap.com/nw04** → *SAP NetWeaver* → *SAP NetWeaver Technical Operations Manual* → *Additional Administration of SAP NetWeaver Components* → *Portal* → *Management* → *Backup/Restore and Recovery*



Planning the Cluster Configuration

Planning the cluster for the SAP Web Application Server helps you to outline the current requirements and growth needs of the portal. In addition, planning eases the installation process, and enhances the chances that the solution you intend to implement matches current and future needs.

In the planning phase, the following factors determine the installation and configuration of the JAVA instances in a cluster:

- **Number of portal nodes to deploy**

The number of portal nodes to be deployed in the cluster relates to the number of JAVA instances. However, consider optimizing JAVA resources on each physical machine.

- **Number of client requests**

The number of dispatcher nodes must increase in proportion to the increase in client requests. Note that only one dispatcher is installed on each physical machine.

For more information on the infrastructure of SAP Web Application Server (JAVA), go to service.sap.com/ti → *Technical Infrastructure* → *SAP NetWeaver* → *Web Infrastructure Concepts for SAP Web Application Server*



Preparing the Portal Cluster

There are several configurations for clustering the portal. However, you must prepare each configuration with the prerequisites and requirements of the portal.

For detailed information on the prerequisites and requirements of the portal, see the guide, *Installation Guide - SAP EP 6.0 (SR1) on WebAS 6.40* at:

service.sap.com/instguidesNW04 → *Installation* → *SAP EP*

To prepare the portal cluster, do the following:

1. Perform all prerequisites for SAPinst and the portal on the portal host. You can dedicate separate machines to both the user persistence store and the database for the SAP Web Application Server (JAVA).
2. Install the JAVA instance of the SAP Web Application Server, which installs the Central Services, and then deploy the portal on the same machine. Although you can install the JAVA instance, and the portal on the same machine as the database, it is recommended to locate them on a machine other than the machines dedicated to the database, and the user persistence store. This portal becomes the development and testing environment.
3. Perform the relevant post installations and configurations for the development and testing environment.
4. Install and configure additional JAVA instances of the SAP Web Application Server on each machine you intend to use in the cluster.



Clustering the Portal Platform

Purpose

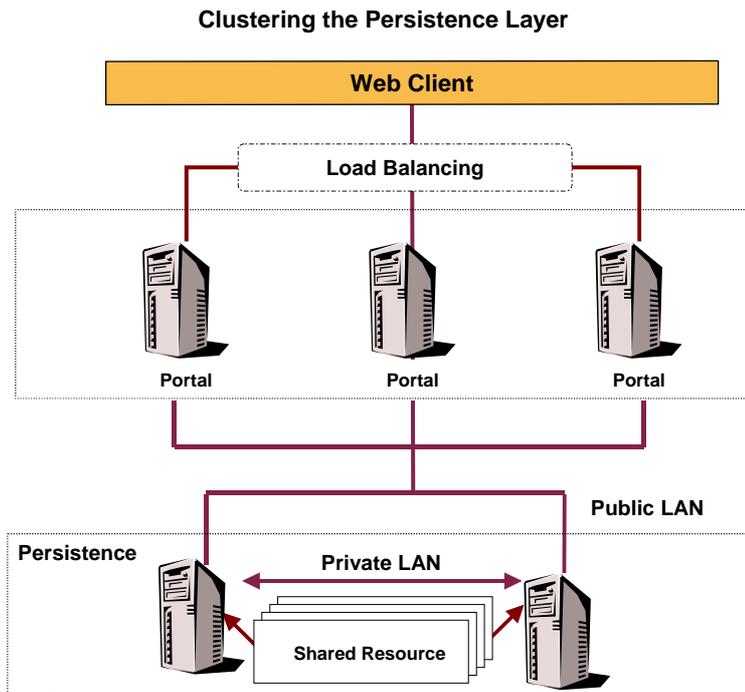
Implementing the persistence layer in a cluster makes the central storage, or shared resources highly available and reliable. Also, when a failure occurs in the persistence layer, resources can be redirected and the workload redistributed to other machines.

Clustering solutions are generally vendor-specific. In your implementation, you need to address the issues of clustering, and involve systems administrators, database system (RDBMS) experts, and Web server administrators.

To identify high availability needs of the portal at the customer's site, consider the following:

- Special hardware requirements of the clustering solution
- Number of users
- Hours of operation
- Projected changes in size and performance requirements
- Portal backup and restore plans

The following illustrates the clustered persistence layer with load balancing:



Identifying and Removing Single Points of Failure (SPOF)

By identifying and eliminating all single points of failure (SPOF) in the portal landscape, you ensure a continuous and highly available portal environment.

Single points of failure occur where only an instance of the software component is deployed for specific tasks in the portal landscape. For example, a single database represents a single point of failure. If that single component fails, there is no alternate one to take its place.

The following represent identified single points of failure in the portal landscape:

- Database

The database is comprised of a schema with data for the Portal Content Directory (PCD), and JAVA applications such as, User Management (UME). The schema contains elements used at runtime in various processes for the portal.

If one database is deployed and used by all the portals in a cluster, it represents a single point of failure. When the database server fails, it shuts down the portal landscape.

Using the specific vendor's recommendation for clustering the database, or other third party solutions, can ensure high availability of the portal.

- Single Points of Failure (SPOF) in the SAP Web Application Server
Detailed information about identifying and removing SPOFs in SAP Web Application Server (JAVA) is available on SAP help portal at: help.sap.com/nw04 → *English* → *SAP Library* → *SAP NetWeaver* → *Application Platform (SAP Web Application Server)* → *Java Technology in SAP Web Application Server* → *Architecture Manual* → *Java Cluster Architecture* → *High Availability and Failover*
- Load Balancing Mechanism
A load-balancing solution allows access to the portal environment, such that all incoming requests are distributed to the JAVA instances according to the load-balancing configuration that is implemented. Therefore making the load-balancing mechanism highly available is recommended.



Load-Balancing the Portal with the SAP Web Application Server

Purpose

You can use the load-balancing features of the SAP Web Application Server to enhance performance of the portal cluster.

On top of the SAP Web Application Server load-balancing features, you can implement a hardware or a software load-balancing solution.

For detailed information on how to apply the load-balancing features of SAP Web Application Server in the portal cluster, go to SAP help portal at: help.sap.com/nw04 → *English* → *SAP Library* → *SAP NetWeaver* → *Application Platform (SAP Web Application Server)* → *Java Technology in SAP Web Application Server* → *Architecture Manual* → *Java Cluster Architecture* → *Load Balancing of the SAP Web AS for Java Applications*



Managing Resources through the SAP Web Application Server (JAVA)

You manage the portal cluster through the SAP Web Application Server, which administers and runs a set of portal servers and their shared resources as though they are on a single machine.

Communication among distributed portal components over several machines is via JAVA instances and the Central Services. A JAVA instance is comprised of a dispatcher (usually one dispatcher for an instance), and one or more server processes.

The Central Services is comprised of the Message Service and the Enqueue Service, which are the basis of communication and synchronization for the cluster.

For detailed information on JAVA cluster architecture, go to SAP help portal at: help.sap.com/nw04 → *English* → *SAP Library* → *SAP NetWeaver* → *Application Platform (SAP Web Application Server)* → *Java Technology in SAP Web Application Server* → *Architecture Manual* → *Java Cluster Architecture*



Performance and Availability in the Cluster

You can make the portal highly available, and enhance its performance using only the JAVA instance of the SAP Web Application Server.

The JAVA cluster includes the following:

- **High Availability and Reliability**

You can vertically scale the server process of J2EE engine in a single machine. In addition, you can implement the same scaling configuration for each physical machine in the cluster. This implementation ensures availability at all times.

When a server process is down, the dispatcher node only redirects client requests to other server processes as the portal is always started on all the processes.

- **Performance and Scalability**

The J2EE Engine is scalable across multiple servers and allows load-balancing, that is, you can add extra machines with additional portal installations to the existing cluster environment, as the use of the portal increases.

For more information on the infrastructure of SAP Web Application Server, go to service.sap.com/ti → *Technical Infrastructure* → *SAP NetWeaver* → *Web Infrastructure Concepts for SAP Web Application Server*

Knowledge Management and Collaboration in the Portal Landscape

When you run the SAPinst installation tool, you can optionally choose to install Knowledge Management and Collaboration. This documentation gives you detailed information on how you can integrate Knowledge Management and Collaboration in the portal landscape. The information is particularly useful, if you have already installed the portal and want to install these components at a later point in time. It focuses on integrating them in a system landscape that is set up with or without load balancing.

Knowledge Management and Collaboration is comprised of the components:

- Content Management and Collaboration
 - Content Management enables documents that are stored in heterogeneous repositories to be managed from a central point, in a consistent manner with the help of a variety of user-centric services.
 - Collaboration provides services that support communication and cooperation on business processes.

For more information, see [Content Management and Collaboration \[Page 16\]](#)

- Search and Classification (TRES)

TRES enables the search and classification operations that are a core part of the Content Management application. If you decide to install Knowledge Management, you have to install both the Content Management and TRES components.

For more information, see [TRES \[Page 22\]](#)



Content Management and Collaboration

Within the system landscape, Content Management & Collaboration can be regarded as a single component because they are always installed together and rely on each other to function properly. The following list identifies the parts of the component, to give you a better understanding of its structure.

Content Management & Collaboration is comprised of:

- The Content Management (CM) application

The application consists of services that are built on top of the Knowledge Management framework. The services provide useful functions for accessing, publishing and managing documents.

- The Collaboration application

The application consists of collaborative functions that are closely integrated with Content Management and also rely on the Knowledge Management framework as the runtime environment. A supplementary component, the **Application Sharing server**, enables synchronous collaboration like the exchange of information in online sessions.

- The database tables

Content Management & Collaboration data is stored in tables of the java database schema. The tables store the configuration data, access control lists for resources and repository metadata. They also store the contents of the standard Knowledge Management repository, the CM repository.

- File system data

A file system repository, the `/etc` repository, stores configuration data for CM in a folder hierarchy. It contains user-defined configuration data as well as customizing data for applications and services. For example, it includes stylesheets for content filters and form-based publishing.

- Configured Repositories

After the installation of CM, various types of repositories can be configured. The contents of these can be stored at different locations, for example, on a file system, on a web server or on a Lotus Notes server. The repository data of a productive Content Management & Collaboration system is therefore generally distributed over several machines.



Content Management and Collaboration Distribution

The Content Management (CM) and Collaboration application is tightly integrated with the portal platform. As a consequence, its location in the portal landscape is dictated by the location of the portal. It must be installed on the same machine as the portal and the database tables have to be located in the same database schema as that of the portal.

In contrast to the above, Search and Classification (TREX) is an autonomous component that does not have to take the location of the portal into account. It is usually set up on one or more dedicated machines. The only requirement is that it establishes a connection to Content Management and the portal during or after the installation.

The optimal location for TREX, CM and Collaboration in the portal landscape depends on the size of the system. The following gives you recommendations for small, medium-sized and large systems. It focuses on:

- Test system
- Production system without load balancing
- Production system with load balancing



Keep in mind that the recommendations here are only general guidelines. An optimal distribution of components is normally worked out in a sizing procedure that carefully analyses the expected workload and takes into account the processing power of available machines.

Test System

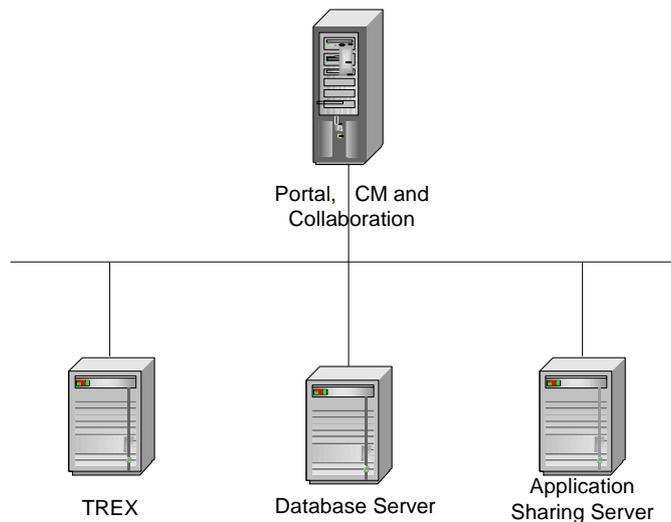
In a small system that is only used for test or demo purposes, the portal platform and Knowledge Management components can run on a single machine. The portal, Content Management and Collaboration, TREX, and the database server can all be installed on the same machine. No distribution of components is necessary.

Production System

In a standard hardware configuration, for a medium-sized production system without load balancing, the distribution of components generally involves several machines. The portal platform, CM and Collaboration are installed on the first machine, the database server on the second and the Application Sharing server on a third machine. The Application sharing server is only required, if users need access to real-time collaboration functions like application sharing.

TREX is normally installed on a dedicated machine that is used exclusively for TREX. In some cases, it can be installed on the same machine as the portal, CM and Collaboration. However, installation on a single machine is only permitted if sufficient memory is available for all the components. With this configuration, a maximum of 6GB memory may be required exclusively for TREX. If insufficient memory is available, this can severely degrade performance.

The graphic shows a system that uses a dedicated machine for the database server, TREX and the Application Sharing Server.



Load-Balanced Production System

A portal platform that has to serve a large number of clients can be set up in a J2EE engine cluster that uses load balancing to distribute client traffic among machines. If the portal is set up in a cluster, CM and Collaboration must also be integrated in the cluster solution. CM and Collaboration are installed on the central JAVA instance and then replicated to each of the machines where the portal is installed. The number of CM and portal instances must be the same. TREX can be installed on a single, dedicated machine or optionally on [several machines \[Page 23\]](#).

In a J2EE cluster, the SAP Web Dispatcher spreads the workload across all the available CM and Collaboration instances which simultaneously process the requests of clients. As a result, the application is able to handle large workloads, avoid performance bottlenecks and grow incrementally with the system load. When the demands of clients increase, new machines with CM and Collaboration instances can be added to the cluster to increase throughput and performance.

If a CM and Collaboration machine fails, the other instances can still continue processing. The dispatcher identifies the failure and automatically redirects requests to the remaining machines. However, the load dispatcher does not support failover. If CM fails on one machine, the processing of client requests is aborted and not automatically transferred to another machine. Also, if the tasks of a service have been scheduled to run on a specific CM instance that fails, they must be rescheduled to run on another instance before processing can continue normally. The cluster can therefore heighten the availability of the system, but not guarantee uninterrupted continuous processing of all requests.

In summary, load balancing in a cluster has the following main advantages. It enables CM and Collaboration to:

- Handle large workloads that cannot be processed effectively by a single machine
- Increase processing capacity incrementally as the workload grows
- Improve availability
- Flexibly change the system landscape to meet requirements



Content Management and Collaboration in a Cluster

This section examines how Content Management (CM) and Collaboration component can be integrated in a portal landscape that is set up as a cluster. It also explains load-balancing mechanisms and the strategies that the application uses to enable several instances to simultaneously process and access the same data.

The main components of CM and Collaboration in a cluster are:

- Several instances of the application
- Tables in the Java schema of the database server

As mentioned earlier, an instance of the application **must** be available on every machine in the cluster where the portal platform is installed. Each instance accesses **the same set of persistent data** that is stored in the database schema. The database schema stores the CM configuration data, resources with their metadata and access control lists.

The CM load in a J2EE Engine cluster is spread across the available machines with the help of two components:

- The SAP Web Dispatcher
- A scheduler service

SAP Web Dispatcher

The SAP Web Dispatcher receives all the requests coming from clients and distributes them to the J2EE instances. It distributes requests in a round-robin procedure, taking the capacity of the individual instances into account. As a result, a new request is always directed to a machine with sufficient capacity. Requests are therefore shared amongst all available machines and the workload is evenly distributed.

Scheduler Service

The CM scheduler service also distributes the load in a cluster. However, it operates independently of the dispatcher. Whereas the dispatcher handles requests coming from clients, the scheduler handles tasks that originate from internal CM services like subscription, indexing and content exchange. It directs the tasks of services to be executed on one or more explicitly specified CM instances. As these tasks can generate a high system load, they should also be optimally distributed.

The scheduler itself does not decide where service tasks are run. After a CM installation, the administrator has to use the settings offered by the scheduler to assign tasks to one or more

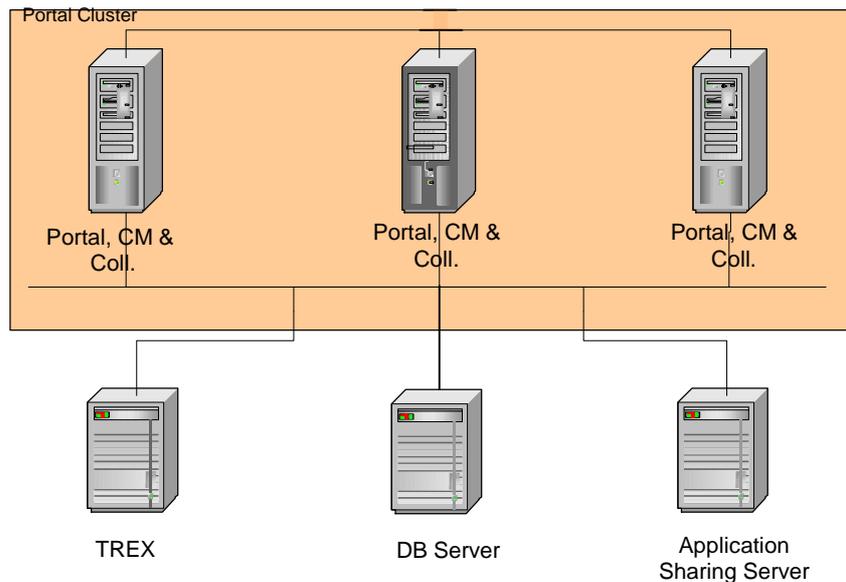
specific CM systems. It is then the responsibility of the scheduler to allocate the tasks to the correct system at runtime.

Integration of Content Management and Collaboration in a Cluster

The following graphics illustrate two different ways in which CM and Collaboration application can be integrated in a load-balanced portal landscape.

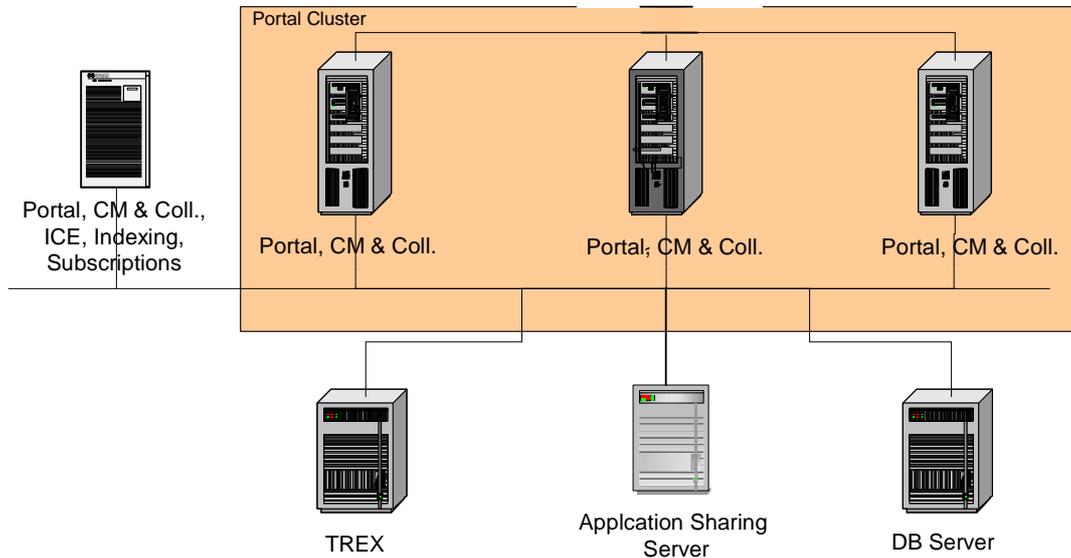
The first graphic shows a standard installation where the location of the application is dictated by the location of the portal platform. An instance of CM and Collaboration runs on each of the machines where a portal is installed. The database server with the Java schema containing the portal, CM and Collaboration tables is located on a machine that is excluded from the load-balancing logic.

All the CM and Collaboration instances access the same database server.



The second graphic shows an alternative setup that includes an additional machine that is excluded from the load-balancing logic. This machine runs a portal, CM and Collaboration instance, but is not used for processing client requests. It is reserved for processing the tasks of CM services that generate a high system load like content exchange, index management and subscriptions. The CM scheduler service ensures that the tasks of the services are directed to the additional machine.

Whether it is advisable to include an extra machine that exclusively handles the tasks of CM services depends on the individual requirements of a system. If, for example, a very high workload is generated by the CM index management service, then the integration of an additional machine would improve performance because it would leave the other machines free to process client requests.



Data Synchronization Strategies

Synchronization of Cache Data

As several instances of the Content Management & Collaboration application access and process the same data, the load balancing solution must ensure that each instance uses current data. In particular, the data cached by each instance must be up-to-date and consistent across all machines. Consistency is ensured with the help of the `cache timeout` parameter for repositories. The parameter specifies how long entries in the local cache of a machine are valid. If entries are no longer valid because the specified time has expired, the application checks whether new data is available on the database. If new data is available, it is read from the database. If not, the existing cache entries are re-used.

Synchronization of Configuration Data

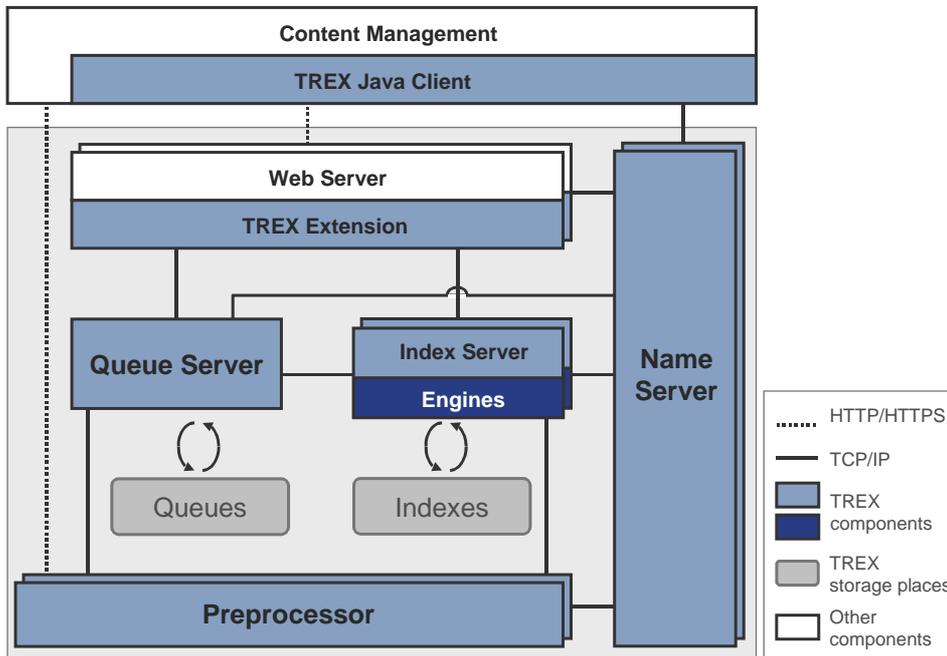
The configuration data is stored in the Java database schema. When data that is significant for all instances is changed, the configuration framework triggers events to inform all instances of the change. Each application running on an instance is responsible for receiving and reacting to such events. If an application registers an event signaling that relevant data has changed then it must read the new data from the database. This strategy ensures that the application always uses the most current data, even if multiple instances are changing the same data.



TREX processes all index, search, and text-mining requests received from the portal. The following are the TREX components:

- Java client
- Web server with TREX extension
- Queue server
- Preprocessor
- Index server
- Name server

The graphic below shows the individual components and how they communicate.



Java Client

TREX provides several interfaces that can be used to integrate TREX functions into an application. The Java client is an interface for Java applications, which is integrated into the SAP J2EE Engine as *TREX Services*. By using the *TREX Services* in the SAP J2EE Engine the TREX functions are available in Content Management and in the portal.

Web Server with TREX Extension

Content Management (more precisely, the Java client) accesses the TREX functions using a Web server. Communication between Content Management and the Web server takes place using HTTP/HTTPS and XML. The Web server receives requests and forwards them to the index server and queue server. The servers then process the requests.

A TREX component that extends the Web server with TREX-specific functions is installed on the web server.

Queue Server

The queue server enables the asynchronous indexing of documents. It has a separate queue for each index. It gathers documents to be indexed into the queues. It transfers documents to the index server for the actual indexing process at regular intervals.

You can use the queue parameters to control when and how many documents are transmitted. This allows you to schedule indexing for times at which the index server does not receive a large amount of search requests.

The queue server forwards the documents to the preprocessor before transmitting them to the index server.

Preprocessor

The preprocessor has two tasks:

- When a search takes place, the preprocessor carries out a linguistic analysis of search queries. The preprocessor passes the results of the analysis to the index server, which then processes the query further.
- When indexing takes place, the preprocessor prepares the documents for the indexing process. The preparation consists of the following steps:

- Loading the document

In the portal, documents are not normally transferred directly to TREX. Instead, they are forwarded in the form of a URI that references the storage location of the document in question. The preprocessor resolves the URI and collects the actual document from the repository.

- Filtering the document

Documents can exist in various formats (Microsoft Word, Microsoft PowerPoint, PDF, and so on). The preprocessor filters the documents, that is, it extracts the text content and converts it to Unicode format UTF 8 for further processing.

- Analyzing the document linguistically

The preprocessor uses a lexicon that analyzes texts in various languages.

Index Server

The index server is responsible for indexing, classifying, and searching. It receives requests and forwards them to the TREX engines. The engines provide the actual core functions of TREX:

- The search engine is responsible for standard search functions such as exact, error-tolerant, linguistic, Boolean, and phrase search.
- The Text-mining engine is responsible for classification, searching for similar documents ('See Also'), the extraction of key words, and so on.
- The attribute engine is responsible for searching in documents properties such as author, creation date, change date, and so on.

Name server

The name server uses its databases to store and coordinate system-wide information. It also ensures that the TREX servers can communicate with each other and that TREX can communicate with Content Management. The name server is also responsible for distributing the system load if more than one TREX server is capable of carrying out a task.



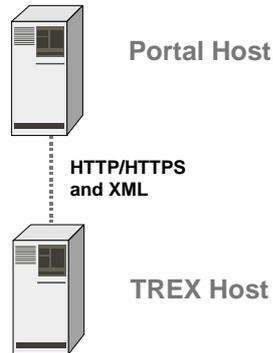
TREX Distribution

TREX offers a flexible architecture and can be adapted to different requirements. You can scale TREX if necessary. Your options range from a minimal system with one host, to a large distributed server landscape.

Single-Host System

A minimal TREX system consists of a single host that provides all TREX functions (indexing, classification, and searching). You can use a minimal system as a demo and test system, or

as a production system. For a production system, we recommend that you install TREX on a dedicated host that is used exclusively for TREX.

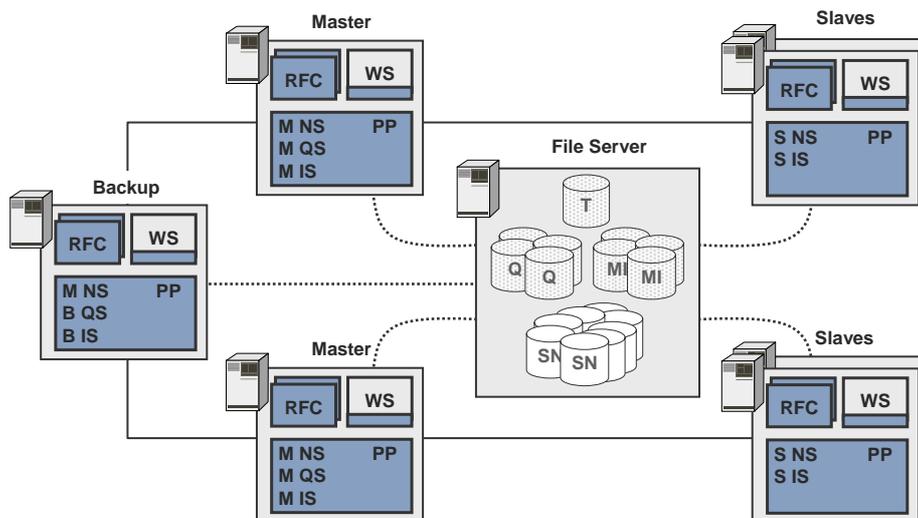


Multiple-Host System

You have numerous options for scaling TREX. You use a scaled scenario to distribute the search and indexing load among several hosts and to ensure the availability of TREX. In a multiple-host system, the individual hosts are responsible for different tasks depending on which TREX components run on them. For example, you can set up dedicated search servers with copies of the original indexes and configure automatic index replication to keep the copies up-to-date.

Example

The graphic below shows an example for a distributed TREX system:



Explanation of abbreviations:

- Master Server: M NS = Master Name Server; M QS = Master Queue Server; M IS = Master Index Server
- Slave-Server: S NS = Slave Name Server; S IS = Slave Index Server;
- Backup-Server: B NS = Backup Name Server; B QS = Backup Queue Server; B IS = Backup Index Server
- Other Servers: RFC = RFC Server; WS = Web Server; PP = Preprocessor
- Data: Q = Queue; MI = Master Index; SI = Slave Index; SN = Index Snapshot; T = Topology File

For details on distribution options and implementation, see the Guide *Distributed Search and Classification (TREX) 6.1 Systems*.

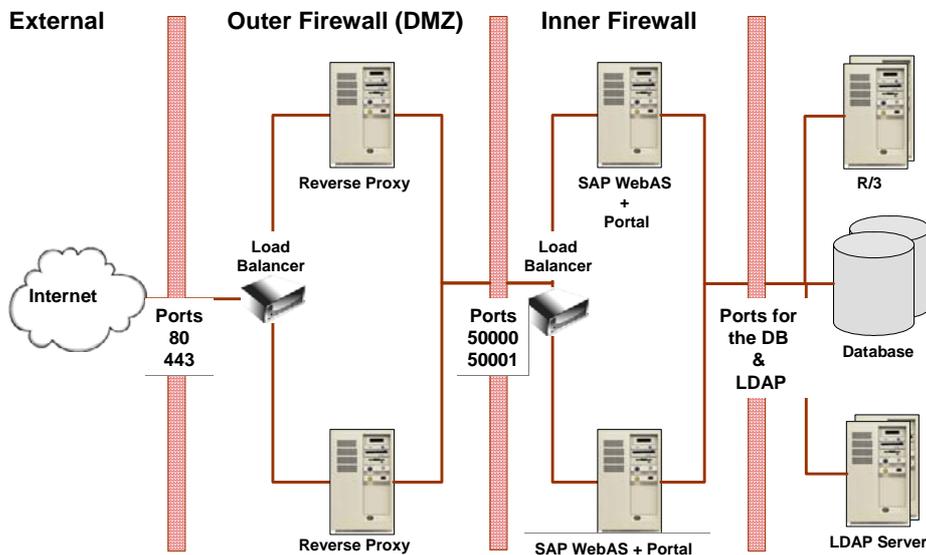
Network Landscapes

When building your portal landscape, keep in mind the following portal requirements:

- The different elements of the persistence layer, such as the database for the SAP Web Application Server and the portal, and the User Persistence Store. These can be distributed among several machines.
- Firewalls must allow access according to the network connections and landscape.
- The portal must have a single HTTP/HTTPS entry point.

Highly available SAP Enterprise Portal can be implemented behind a firewall to provide increased security while processing client requests. In this case, you deploy a reverse proxy server that obtains requests from portal clients and forwards them to the portal. Responses from the portal also go through the reverse proxy server, which delivers them to the clients.

The following figure illustrates the production environment of a highly available SAP Enterprise Portal in a firewall:



The administrator must configure the firewall in accordance with the necessary free ports for the corresponding services, such as HTTP(s), and so on.

The following are the ports to open in the inner firewall:

Component	Service Name	Port	Protocol
LDAP directory service The connection between the portal and the LDAP directory service can also use SSL..	Lightweight Directory Access Protocol	389 Default SSL port is 636	TCP
Microsoft SQL Server	MSSQL Server	1433 For SSL you may use 1434	TCP
Oracle database server	Listener	1521 For SSL you may use 1527	Oracle service
Web Publishing – HTTP	HTTP	80	HTTP

Web Publishing –HTTPS	HTTPS	443	HTTPS
Portal - HTTP	HTTP	50000	HTTP
Portal - HTTPs	HTTPs	50001	HTTPs

Note that these are the default port numbers used for the specified component, however, they can be changed according to the customers' network topology. If a port cannot be opened for a component in the inner firewall, that component can be moved to the DMZ as well.

Portal in the DMZ

Purpose

In this landscape, the portal runtime resides in the DMZ and receives all the HTTP/HTTPS requests from the Internet clients. It then processes the requests and returns the results to the clients. There is a direct connection between the portal server and the clients through a single HTTP/HTTPS port in the firewall.

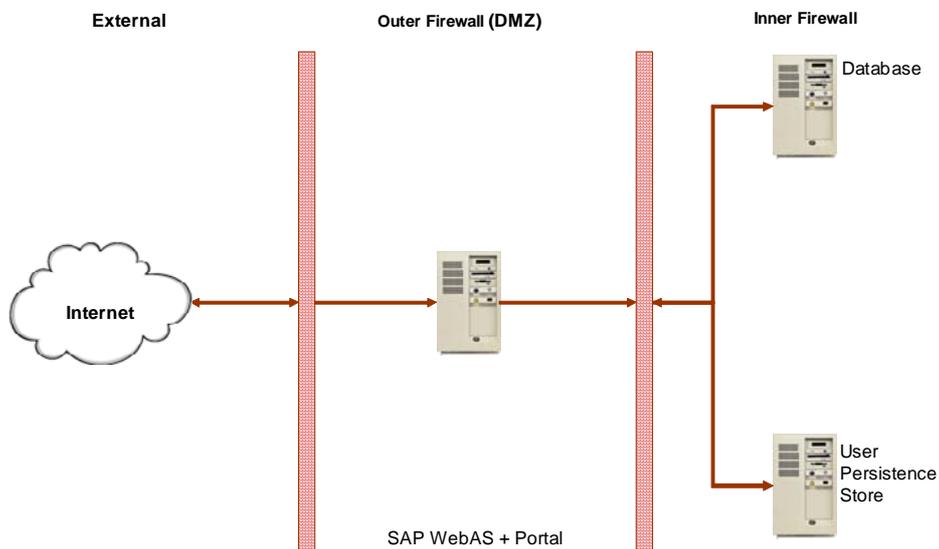
The persistence layer resides in the secured network. It is possible to move some components from the persistence layer to the DMZ according to the security and backup policy of the customer.

In this landscape, the inner firewall separates the portal runtime and its persistence layer. The following are the requirements of the inner firewall settings:

- LDAP directory port.
- If there are connections to back-end applications, the portal needs to access them through the firewall.

If these applications have a Web interface in the DMZ, such as the Internet Transaction Server (ITS), the connection is to this interface.

The following illustration shows the portal platform in a firewall configuration:



If a port cannot be opened for a resource in the inner firewall, that resource can be moved to the DMZ as well.



System and Portal Administration

In any infrastructure, there is the need to maintain the interoperability of the various systems, such as security administration, systems administration, and application administration.

Some portal operations such as URL redirection, firewall configuration, Domain Naming System (DNS), reverse proxy settings, and specifically, the configuration of the servers for the load-balancing system all influence the infrastructure. Hence, troubleshooting requirements and responsibilities among the various systems such as network, system, and application administration must be clearly defined.

For more information on the portal administration, go to SAP help portal at:

help.sap.com/nw04 → *English* → *SAP Library* → *SAP NetWeaver* → *People Integration* → *Portal* → *Administration Guide*



Security

Security is a primary concern for enterprises as the portal becomes highly available. Access to most sources of information requires end users to be authenticated. Since information sources may contain sensitive data, it is important that users can be identified and their identity authenticated.

Generally, users access the portal with security credentials configured for the local computer. Using portal tools, you can configure user-related data for logging onto the portal. From the portal, users can access applications, data and other services to which they have permissions using single sign-on, when they are successfully authenticated.

For additional information about the portal security, refer to the security guide for the portal on SAP help portal at: **help.sap.com/nw04** → *English* → *SAP Library* → *SAP NetWeaver* → *Security* → *SAP NetWeaver Security Guide* → *Security Guides for the SAP NetWeaver Products* → *Portal Security Guide*