Lotus. software

IBM WebSphere Portal V6 Self Help Guide

Key recommendations for optimal configuration and use

Problem avoidance, determination, and resolution

Best practices for security and maintenance

Philip Monson
Fang Feng
Jerry Dancy
Shadi Albouyeh
Chakravarthy Kunapareddy
Stephanie Martin
James Roca
John Chambers

Redpaper

ibm.com/redbooks

International Technical Support Organization

IBM WebSphere Portal V6 Self Help Guide

January 2008

REDP-4339-00

Note: Before using this information and the product it supports, read the information in “Notices” on page vii.

First Edition (January 2008)

This edition applies to IBM WebSphere Portal Version 6.

Contents

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:
IBM Director of Licensing, IBM Corporation, North Castle Drive, Armonk, NY 10504-1785 U.S.A.

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

Trademarks

The following terms are trademarks of the International Business Machines Corporation in the United States, other countries, or both:

AIX 5L™
AIX®
Cloudscape™
developerWorks®
Domino®
DB2®
Electronic Service Agent™
HACMP™
i5/OS®
IBM®
Lotus®
OS/390®
Passport Advantage®
pSeries®
Rational®
Redbooks®
Redbooks (logo) ®
RDN™
System i5™
System x™
System z™
Tivoli®
WebSphere®
Workplace™
Workplace Web Content Management™
z/OS®

The following terms are trademarks of other companies:

Oracle, JD Edwards, PeopleSoft, Siebel, and TopLink are registered trademarks of Oracle Corporation and/or its affiliates.

Enterprise JavaBeans, EJB, Java, JavaBeans, JavaScript, JDBC, JMX, JNI, JSP, JVM, J2EE, Solaris, Sun, and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Active Directory, Internet Explorer, Microsoft, SQL Server, Windows, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Linux is a trademark of Linus Torvalds in the United States, other countries, or both.

Other company, product, or service names may be trademarks or service marks of others.

Preface

This IBM® Redpaper focuses on considerations for the optimal configuration and use of IBM WebSphere® Portal Server. We provide you with the information you need to deploy and manage your WebSphere Portal infrastructure, with the goal of problem avoidance. However, if issues occur, the reader is introduced to the various tools and techniques for problem determination and problem solving, including obtaining and installing fixes, how to contact support, and what type of information you should provide before engagement.

This guide is a must have resource for IT architects and administrators throughout the life cycle of a WebSphere Portal environment, from conception and planning to use and maintenance

The team that wrote this Redpaper

This Redpaper was produced by a team of specialists from around the world working at the International Technical Support Organization, Cambridge, MA.Center.

Philip Monson is a Project Leader at the ITSO Lotus® Center in Cambridge MA. Phil has been with Lotus / IBM for 17 years, joining the company when the early versions of Notes were rolled out for internal use only. He has served in management, technical, and consulting roles in the IT, Sales, and Development organizations.

Fang Feng is an Advisory Software Engineer in the IBM Software Group. He joined the WebSphere Portal Level 2 support team in Research Triangle Park, North Carolina, in 2002. His areas of expertise include Portal security, system administration, WebSphere Member Manager, and XMLaccess. He has been working with IBM for 11 years. He holds a Doctor of Philosophy in Computer Science from Texas A&M University.

Jerry Dancy is a Senior IT Specialist who works as a Technical Lead for the WebSphere Portal Server Level 2 Support team. He has five years of experience in WebSphere Portal Server support and previously worked as an Oracle® DBA for four years. He holds a degree in Accounting and CIS from Appalachian State University. His areas of expertise include installation, upgrading, configuration, and clustering of WebSphere Portal. He also has worked on and has led many projects to improve WebSphere Portal Server serviceability. He has written extensively on WebSphere Portal Server installation, configuration, and clustering. Jerry is also an author of the IBM Redbooks® publications *WebSphere Portal V5.0 Production Deployment and Operations Guide*, SG24-6391 and *WebSphere Portal Version 6 Enterprise Scale Deployment Best Practices*, SG24-7387.

Shadi Albouyeh is an experienced WebSphere Portal Software Engineer. She has been working in WebSphere Portal support for over four years since graduating with a B.S degree in Computer Science from North Carolina State University (Raleigh). She is currently the Team Lead of the Portal-Install L2 support team and has previously worked on the WebSphere Portal-API L2 support team. She focuses now on the WebSphere Portal Installation and Configuration aspect of the product, supporting customers with installation and configuration, clustering, enabling security, database transfer, Fix Pack installs, and upgrades.

Chakravarthy Kunapareddy is a Senior technical consultant and an IBM certified professional working with Ascendant Technology (http://www.atech.com), a premier IBM Business Partner. He has over six years of consulting experience with the IBM suite of products of WebSphere Portal, WebSphere Application Server, Tivoli® Access Manager, DB2®, and WebContent Management. He is an experienced infrastructure consultant with expertise in planning, architecture, installation, configuration, deployment, and troubleshooting. He holds a Bachelors Degree in Computer Science and Engineering from Bharathidasan University, India.

Stephanie Martin is a Systems Integration Professional in IBM Integrated Technology Division. Since joining IBM in 2001, she has worked in the IBM Early Deployment Center (EDC), designing and implementing beta, proof of concept, and enterprise scale solutions of Lotus software offerings. She currently acts as the EDC's Infrastructure and Administration lead for the award-winning IBM Workplace™ for Customer Support Portal.

James Roca is a Senior Consulting IT Architect with the IBM Software Group. He has spent the last two and a half years assigned to the Asia Pacific region to build and promote technical skills, and to champion leading edge Portal architectures. Previously, James worked at the IBM China Software Development Lab and the IBM Hursley Development Lab, in the capacity of IT Architect and Solution Consultant. He jointly developed the Portal Perform guide for the IBM EMEA geography. He is also credited with developing the Portal Build & Validate method, which, when adopted, minimizes implementation failure. Most recently, James took over as the Leader at Large of the IBM Worldwide Portal Community. James previously co-authored the *WebSphere V3.5 Handbook*, SG24-6161 and *IBM WebSphere V4.0 Advanced Edition Security*, SG24-6520.

John Chambers is a Knowledge Engineer for IBM WebSphere Portal support in the US. He has been supporting WebSphere Portal for more than six years and is currently focusing on improving the quality of support content, self-help information, and tools available to customers. John has been with IBM support for 12 years, since receiving his degree in Geology from Guilford College in North Carolina.

Thanks to the following people for their contributions to this project:

Thomas Hurek, WebSphere Portal Chief Programmer Fix Packs and Architectural Lead L3, IBM Software Group

William Trotman, WebSphere Portal L2 Support, IBM Software Group

Lauren Wendel, Product Manager - WebSphere Portal, IBM Software Group

Flemming T Christensen, Technical Quality Champion - Lotus, IBM Software Group

Walter Haenel, Portal Architect for Deployment and Operations, IBM Software Group

Yen Li Yong, IBM Software Services, IBM Software Group, IBM Malaysia.

Brett Gordon, WebSphere Portal L2 Support, IBM Software Group

Become a published author

Join us for a two- to six-week residency program! Help write a book dealing with specific products or solutions, while getting hands-on experience with leading-edge technologies. You will have the opportunity to team with IBM technical professionals, Business Partners, and Clients.

Your efforts will help increase product acceptance and customer satisfaction. As a bonus, you will develop a network of contacts in IBM development labs, and increase your productivity and marketability.

Find out more about the residency program, browse the residency index, and apply online at:

ibm.com/redbooks/residencies.html

Comments welcome

Your comments are important to us!

We want our papers to be as helpful as possible. Send us your comments about this paper or other IBM Redbooks publications in one of the following ways:

- Use the online **Contact us** review form found at:

  ibm.com/redbooks

- Send your comments in an e-mail to:

  redbooks@us.ibm.com

- Mail your comments to:

  IBM Corporation, International Technical Support Organization
  Dept. HYTD Mail Station P099
  2455 South Road
  Poughkeepsie, NY 12601-5400

1

Introduction

This chapter provides you with an overview of this Redpaper, highlights some of the new features in IBM WebSphere Portal Version 6, and provides a general description of what will be covered in each chapter.

1.1 Purpose of this Redpaper

The WebSphere Portal Self Help Guide focuses on the who, what, where, when, and why of a WebSphere Portal Server Version 6 deployment. The goal of this guide is to introduce and explain the various scenarios that you should consider before, during, and after the installation of WebSphere Portal Server. Our mission as authors is to arm you with the conceptual information that you need to deploy and manage your portal infrastructure, with the goal of problem avoidance. We do recognize that even in the best of circumstances problems can occur, so we also introduce you to various tools and techniques for problem determination and problem solving, including obtaining and installing fixes, how to contact support, and what type of information you should provide before engagement.

For the purposes of this Redpaper, our concentration is focused on the underlying framework that composes the WebSphere Portal Server core (that is, access control, integration, administration, and presentation). Our goal is to assist you in answering such questions as:

- What tools can I use to obtain sizing estimates for my portal environment?
- When/Why should I convert my portal server(s) from Cloudscape™ to an external database?
- What can I do to optimize the runtime in my portal environment?
- How do I convert my portal server(s) from a test LDAP to a production LDAP?
- Why are dual lines of production important in a portal cluster?

For customers looking to leverage the additional services and features that WebSphere Portal offers, we do provide information about recommended supplemental materials that help make up WebSphere Portal's portfolio, such as WebSphere Content Management, Portal Document Manager, and Personalization, among others.

For customers who are looking to install and configure a enterprise deployment of WebSphere Portal Server Version 6, refer to the IBM Redbooks publication, *WebSphere Portal Version 6 Enterprise Scale Deployment Best Practices*, SG24-7387.

For existing customers looking for a step-by-step guide to migrate their WebSphere Portal Server environment from Version 5.1 to Version 6, we encourage you to read the Redpaper *IBM WebSphere Portal V6: Best Practices for Migrating from V5.1*, REDP-4227.

This Redpaper is oriented toward Portal Administrators and IT Architects at all levels of administration and architectural design. A working knowledge of J2EE™ Architecture and WebSphere Application Server administration, as well as a basic understanding of Java™ programming concepts, are assumed.

1.2 IBM WebSphere Portal Server overview

Figure 1-1 shows an overview of IBM accelerators for WebSphere Portal.

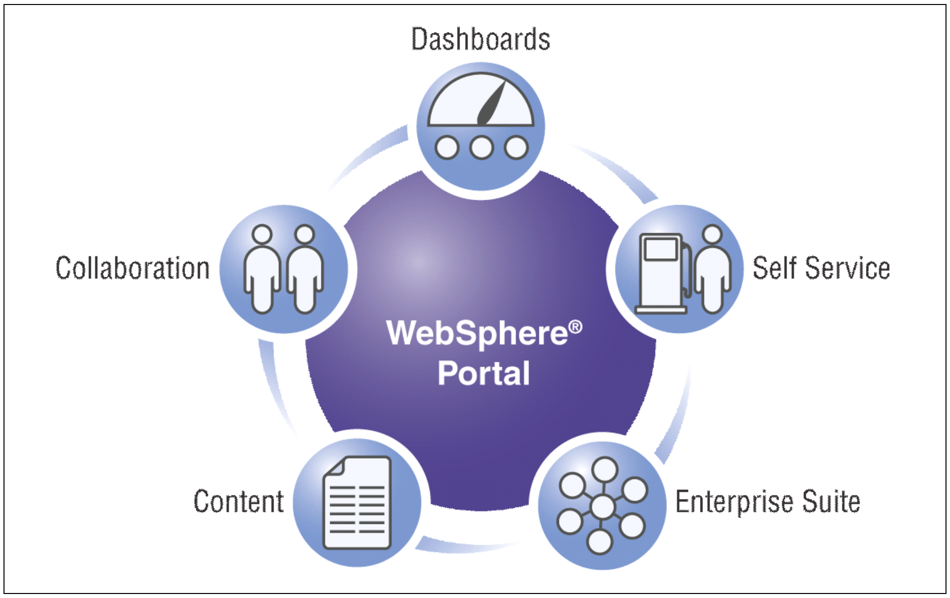


Figure 1-1 IBM Accelerators for WebSphere Portal

IBM WebSphere Portal Version 6 is an enterprise portal solution with the complete portal services that are necessary to deliver a single point of personalized interaction to applications, content, business processes, and people for a unified user experience. WebSphere Portal improves overall productivity and customer satisfaction. WebSphere Portal provides for improved operational efficiency and productivity, as well as accelerated application and content deployment, by integrating some of the best technology that IBM has to offer. WebSphere Portal is a responsive and reliable portal platform, with a wealth of solutions available to address the needs of your On Demand Business, all from a recognized leader in the enterprise portal market.

1.3 What is new in WebSphere Portal Version 6

Figure 1-2 shows an example of a business portal solution.

Home

Product News

IBM WebSphere Portal delivers power and flexibility again!

What's New?

Get Started.

Write Something

Home

Existing Customer?

Need Help?

Go to the IBM Infocenter for complete and in depth product information.
Also, try IBM Developer Works for technical articles about various topics related to IBM WebSphere Portal.

Support?

Go to the the IBM e-support site. Or, try out IBM Newsgroups.

Product Links

Administration
Web Content Management
Document Management
Mail
Calendar

Other Links

IBM.com
IBM software home page
IBM WebSphere
IBM Workplace
Partner World
Linux at IBM

Figure 1-2 Example of business portal solution

IBM WebSphere Portal Version 6.0 delivers new features, functions, and performance that helps to improve the efficiency of your organization, the speed of your application deployment, and the utilization of your IT assets.

Some of the new features in Version 6 include:

- A new user interface featuring AJAX and drag and drop customizations to help portal users accomplish more with fewer clicks.
- Portal Applications can be enhanced with orchestrated flow and electronic forms services that allow employees, partners, and customers to execute transactions faster.
- Application templating and easier portlet development accelerates application deployment and customization through the innovative use of services-oriented architecture (SOA).
- Inline content editing and powerful personalization help increase employee productivity and customer satisfaction.
- Intuitive administration tools and performance improvements deliver responsiveness and reliability at lower cost.
- A set of add-on business-ready packages or “accelerators” that support a quicker ROI and shorter implementation for specific business problems.
- Improved operational efficiency and productivity by linking the right people, process, and information so transactions are executed quickly and accurately.
- Accelerated application and content deployment through new tools and the innovative use of services oriented architecture (SOA).
- Lower overall cost of portal deployment with faster performance and easier administration.

- Responsiveness and reliability, delivered by a leader in the enterprise portal market.

1.4 Administration improvements

There are a number of enhancements and new features in Version 6 that are central to administration. Some of the highlights include:

- Portal configuration management integrated with WebSphere Application Server configuration management for easier operation of clustered portal installation, less manual steps, and reduced risk for failure.
- Attribute Based Administration: Provides the option Use Personalization Rules to show or hide pages and portlets based on user attributes. Visibility Rules allows administrators the option to display Web content based on any type of information, including LDAP attributes, time of day, or session information.
- Multiple LDAP support: Realms can now be pointed to one user registry or multiple user registries, reducing the need for investing and implementing a directory consolidation solution.
- Data Domains: Portal now allows the separation of portal data into multiple domains. Domains can be shared across multiple independent lines of production, aligning with the 24/7 requirements of an enterprise scale deployment.
- Web Content Management Clustering: Multiple nodes can share the same repository (JCR), providing a single point of administration for multiple WCM servers.
- Policies: Simple management by assigning policies to control the behavior of a resource.
- Domino® and Extended products: Configuration of Domino and Extended Products and portlets is now automated by the Domino-Portal Integration Wizard, saving manual steps for the administrator.
- WebSphere Process Server support for most platforms: As of WebSphere Process Server Version 6.0.2, remote connectivity through the process server client can now be done from WebSphere Portal Server; also, single server installations of process portal can now be federated and clustered.

1.5 Structure of the Redpaper

Figure 1-3 gives an overview of the structure of this Redpaper.

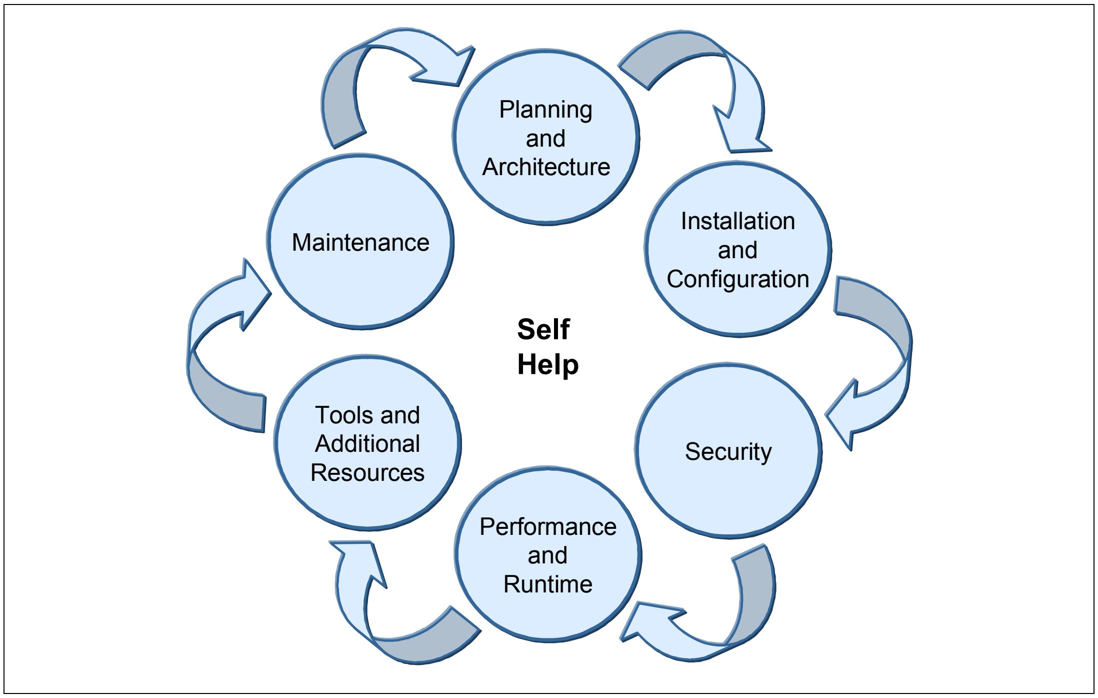

Planning and Architecture
Installation and Configuration
Security
Performance and Runtime
Tools and Additional Resources
Maintenance
Self Help

Figure 1-3 Structure of this guide

This section describes how the Redpaper was constructed and provides a summary of the information that is contained within the chapters.

Regardless of the complexity of your deployment, the greatest factor in how successful a customer solution will be is how well it is planned. In Chapter 2, “Architecture and planning” on page 9, we provide a roadmap discuss the planning and design of your WebSphere Portal environment. Project Managers, Business Sponsors, Software Developers, and other key stakeholders in your organization are strongly encouraged to review the material covered here.

Chapter 3, “WebSphere Portal installation” on page 55” covers the installation and configuration of WebSphere Portal Server using the flexible deployment options for the most common topologies.

WebSphere Portal Server provides a number of mechanisms to help keep your assets protected. In Chapter 4, “WebSphere Portal security” on page 85, we discuss the different components in WebSphere Portal that provide the security features and how they can be integrated into your infrastructure to provide a secure solution.

A Web portal is an integrated solution that requires the collaboration of many teams to implement. Any low-peforming part of the integrated solution can cause overall portal performance degradation. In Chapter 5, “WebSphere Portal runtime and services” on page 137”, we discuss how topology, application design, back- and front-end resources, and other factors can greatly impact the user experience and provide information about monitoring tools that can help to prevent bottlenecks.

Functional challenges, can affect even the best thought out and executed deployments. In Appendix A, “Using IBM tools to find solutions and promote customer self-help” on page 169, we discuss the usage of the various support tools to enable customers to self- recover from operational challenges more quickly.

Appendix B, “Maintenance: Fix strategy, backup strategy, and migration strategy” on page 207 is broken up into three parts: maintenance, backup, and migration. For the maintenance portion, we show you how staying current with the latest maintenance releases to leverage the included fixes to preventively fixes issues, and when to switch to later releases to introduce additional features. Performing regular backups is the surest way to protect your systems and critical data from loss due to hardware/software failure. The information and guidelines presented in Appendix B, “Maintenance: Fix strategy, backup strategy, and migration strategy” on page 207 on backup strategy are provided to help you to understand your backup software and hardware options, and encourage you to perform system backups regularly. Finally, in the Migration section, we briefly discuss the migration path for WebSphere Portal Server Version 5.1 customers looking to migrate to WebSphere Portal Server Version 6 and the tools and resources available to help you in planning and implementation.

2

Architecture and planning

IBM understands and recognizes that many customers need to make important decisions about their WebSphere Portal Server solution, both prior to and during a deployment.

With intimate knowledge of the challenges and pitfalls that go hand in hand with managing many large scale WebSphere Portal deployments, this chapter sets out to provide the reader with an informed approach to planning, architecting, and implementing a successful Portal deployment.

Although this chapter is written with a bias towards enterprise scale WebSphere Portal Server deployments, the principles nevertheless remain applicable to smaller deployments.

2.1 Building the right Portal architecture

WebSphere Portal Server architectures come in many shapes and forms. This is in part attributed to the demands of modern day e-business, where the need to establish a robust, open, scalable, and strategic infrastructure platform to set the standard for system reuse and interoperability exists. WebSphere Portal Server achieves all of these goals through the establishment of an extensible framework of services, and then, by being deployed as an Enterprise Application on top of either WebSphere Application Server or WebSphere Process Server (certain restrictions apply).

Leveraging upon the strengths of the underlying WebSphere technology make it possible for WebSphere Portal Server to support everything from the small workgroup (WebSphere Portal Express) to the high-volume enterprise, to the geographically distributed Portal. Indeed, IBM recognizes that “one size does not fit all” when it comes to planning and architecting a Portal solution.

To the experienced Portal Solution team, functional and operation aspects need to be considered with equal rigor and importance when implementing a successful Portal solution.

It is acknowledged that the principles of good architectural design and development go hand in hand with the adoption of a suitable methodology. Indeed, the IBM Global Services Method (GS-Method or GSM) has been the basis for many successful WebSphere Portal Server deployments. However, the merits and application of such methodologies are beyond the scope of this particular Redpaper.

2.1.1 Addressing functionality

Functionality remains the main driving force behind many of the systems and solutions that we use each day. Without functionality, a system or solution fast becomes obsolete. Portals are no exception to this rule.

Although no specific functional requirements are documented within this chapter, as attempting to do so would prove futile given the uniqueness of many WebSphere Portal Server deployments, one can dramatically improve the success factor of a deployment by accurately capturing conditions such as the following:

- What the specific applications, services, or products are that a WebSphere Portal Server implementation should support.
- What the high level capabilities are that an implementation should have, for example, security, user collaboration, user interface, and so on.
- What the general use cases are that best describe the business functionality required from the implementation.

2.1.2 Addressing integration

Integration is not a trivial issue and requires time and effort to accurately establish the most appropriate solution. A good WebSphere Portal Server architecture, therefore, addresses integration as early on in a project as possible.

WebSphere Portal Server integration can be loosely classified into the following three categories.

Presentation Integration

This integration approach represents the simplest method of incorporating content into a WebSphere Portal Server deployment and is based solely upon the ability to screen scrape, either through the deployment of an iFrame or Web Clipping Portlet, existing visual content served by one or more back-end servers. This approach, however, has the severe drawback that content cannot be personalized or manipulated in any shape or form. Furthermore, in terms of overall performance, presentation integration does not normally sit well with enterprise scale deployments due to the lack of any type of brokerage or connection pooling mechanism for reducing the amount of back-end requests. For example, a Portal page containing two iFrame portlets will result in two separate back-end calls for a single Portal page request. This is irrespective of whether the content is served by the same back-end server or not.

Application or Programmatic Integration

At a high-level, Application or Programmatic Integration provides for the very dexterity that Presentation Integration does not. Furthermore, because Application or Programmatic Integration lets you represent information in whatever shape or form is most appropriate to your target audience, it is the perfect solution for most implementations. The key to achieving this is the ability to dictate, through a custom coding effort, what happens to the actual data of a request. This extends to both the presentation and business logic aspects of an application, for which the Model-View-Controller (MVC) pattern is arguably the most well known programming concept. One drawback with this integration approach is the amount of effort required to create such custom developed components. This can be particularly challenging when an organization’s core business is other than software development. Indeed, most organizations can no longer afford the time or the cost of development to write new applications each time their business requirements change. Instead, they prefer to purchase Commercial-Off-The-Shelf (COTS) portlets or to use wizard driven development, as found in IBM Rational® Application Developer and IBM Portlet Factory.

Middleware Integration

In a subtle distinction from Application or Programmatic Integration, Middleware Integration commonly involves the deployment of an intermediary. Such an intermediary may perform queuing, routing, transformation, workflow, or even business choreography. In addition, an intermediary maybe used to attain a specific QoS (Quality of Service) or to provide a layer of abstraction between participants in an implementation. Middleware can also be used to bridge the gap between different technologies, standards, and even vendors.

2.1.3 Technology choices for connectivity

When considering the various types of integration applicable to a WebSphere Portal Server deployment, it is also often helpful to understand which type of connectivity best suits the actual approach. It is also worth remembering that non-technical factors, such as available skill set and standards within an organization, may influence the choice of a particular type of connectivity.

Web Services

Web Services are based on an open-standards way of defining and invoking a service. The implementation of the requestor and provider are hidden from each other, allowing portability in implementation. The coupling is based on the service interface and a variety of transport protocols can be used. Both synchronous and asynchronous communication is possible, but each service defines the mode it supports. The basic stack is comprised of HTTP, XML, SOAP, WSDL, UDDI, and WSFL. Web Services can employ XML as an encoding schema that is widely adopted. They are relatively “heavy” to implement, and are best suited to inter-enterprise communication, or adopted as an enterprise wide standard for leveraging an ESB, for example. Web services are not built to be high performing, so are not suitable for transactions that require very large throughput.

Messaging

Messaging interfaces such as WebSphere MQ and JMS are based on the asynchronous exchange of messages between producers and consumers. Point-to-Point and Publish-Subscribe communication patterns are provided. Messages are placed on a queue by the sending application, and those messages are then consumed by a receiving application. With messaging, you take advantage of a simple and common API. You adopt industry-standard programming models and you make these available on a selection of operating systems. Messaging provides assured delivery for business critical information. Messaging provides asynchronous (as well as synchronous) processing for loose coupling of applications and control of the rate at which information is processed.

Adapters

Adapters provide access to business logic in a tightly coupled manner. An adapter is specific to a particular Enterprise Information System (EIS) and generally requires client code to be written to parse the proprietary format of the data provided by the EIS. However, this tight coupling allows an adapter to map security, transaction information, and other Quality of Service information between the client and the EIS based on the well-established capabilities of EIS gateways. While adapters typically provide a synchronous interface, the latest specifications define an asynchronous mode as well, and some adapters implement this mode.

Table 2-1 gives you some comparisons between the connectivity types.

Table 2-1 Connectivity comparisons

	Web Services	**Messaging**	**Adapters**
Interface Coupling	Tight.	No. An application may process a variety of messages.	Tight.
Transport Coupling	Loose.	Tight.	Tight.
Implementation Portability	Yes.	Yes.	No.
Security	Standards defined - Not universally implemented.	Vendor-specific.	EIS-specific.
Transaction Support	Standards defined - Not universally implemented.	Limited in scope to queue entry point.	Yes.
Synchronous Invocation	Yes.	Custom implementation.	No.
Asynchronous Invocation	Yes.	Yes.	EIS-specific.
Event Driven	Yes.	Yes.	EIS-specific.
Reliable Payload Delivery	Standard Defined.	Yes.	EIS-specific - Functionality provided by actual adapters.

The following recommendations are made with regards to the selection of the most appropriate connectivity technology:

- Use Web Services when portability or interface standardization is a prime concern.
- Use Messaging when high QoS constraints and loose coupling or asynchronous invocation is needed.
- Use JCA when high QoS constraints and synchronous invocation are needed.

2.1.4 The System Context Diagram

If there is one diagram that can simplistically represent a WebSphere Portal Server implementation, or for that matter any other generic software implementation, then it is the System Context Diagram, as shown in Figure 2-1.

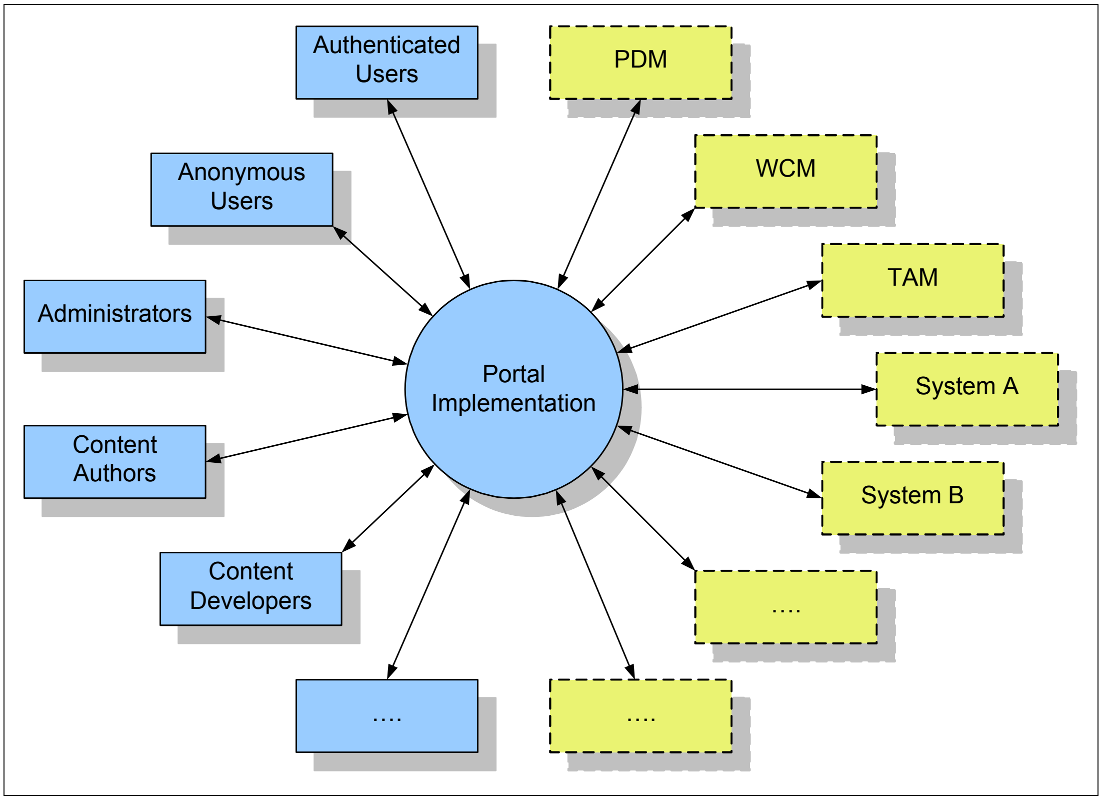


Figure 2-1 System Context Diagram

Figure 2-1 illustrates the various system components and most significant roles of the system. Besides that, it helps to identify in high level terms the systems to which a deployment interfaces. Table 2-2 further explains the various system components and roles.

Table 2-2 System Context Diagram

System Component / Roles	Description
Anonymous Users	An Anonymous User has access to the limited external Portal pages, but never signs into the Portal. Anonymous users can become authenticated users by logging in.
Authenticated Users	An Authenticated User is a user that has logged into the Portal during their current user session.

System Component / Roles	Description
Administrators	Administrators are responsible for the management of the Portal. They are responsible for adding new portlets, new pages, new administrative users, and so on.
Content Developers	Content Developers are responsible for creating Web Content Management (WCM) design artifacts, such as site frameworks, and authoring and presentation templates.
Content Authors	Content Authors are a subset of Authenticated Users that are delegated the responsibility of creating WCM content.
Content Approver	Content Approvers are a subset of Authenticated Users that are delegated the responsibility of approving WCM content. Such users approve or reject content prior to releasing the content for delivery.
PDM	Portal Document Manager (PDM) is a subcomponent of WebSphere Portal Server responsible for archiving and managing documents.
WCM	Web Content Management (WCM) is a subcomponent of WebSphere Portal Server responsible for the complete life cycle of Web content information.
TAM	Tivoli Access Manager (TAM) is an External Security Manager responsible for providing enterprise wide security.
System A	System A is responsible for function X.
System B	System B is responsible for function Y.

2.1.5 Addressing non-functional requirements

Capturing the non-functional requirements is a preliminary task that not only provides a starting point for selecting and sizing the physical components of a Portal solution, but also establishes such key aspects as availability, backup and recovery, disaster recovery, and systems management. In terms of the former aspects, a resulting sizing estimate is normally calculated based on those non-functional requirements giving an approximation of the physical resources required to support the proposed implementation. Of course, many factors influence the selection of the physical resources and actual experiences will vary from that of the sizing estimate for many reasons; the degree of variability can range from the small to the very significant. For those latter aspects, which by no means are exhaustive, the required solution characteristics and capabilities take shape and drive the selection of the hardware and software technologies needed to deliver the proposed implementation, within the constraints of technology, skills, and budget.

Unfortunately, the non-functional requirements of a solution also tend to be treated as "second-class citizens" because they do not add any new or improved functionality. Thus, they typically do not receive the proper attention of executives, the project manager, or even the technical team. However, a project must address the likes of availability and performance in all phases of a project life cycle to be successful.

For those customers finding themselves in the unfortunate situation of having selected and purchased “bare metal: systems, without having undertaken a thorough non-functional requirements study, the degree of usefulness attributed to fully capturing the non-functional requirements at a later stage is somewhat limited. There remain a number of key objectives that the implementation should strive to meet.

The following non-functional requirements are documented to articulate the critical elements of a successful implementation:

- Availability
- Backup and Recovery
- Capacity Estimates and Planning
- Disaster Recovery
- Extensibility/Flexibility
- Failure Management
- Performance
- Scalability
- Security
- Service Level Agreements
- Standards
- System Management
- Usability

> **Tip:** A non-functional requirement is not well specified if it is not specific or measurable. Attainability and measurability are checks that should be performed against each requirement. A requirement should only be included if it is attainable and realizable.

2.1.6 Frequently asked questions about sizing

The most frequently asked questions in terms of non-functional requirements are typically those regarding sizing or capacity planning. For example, given a specific Portal deployment and an anticipated traffic load, what kind of configuration will satisfy the sizing requirements? For example, Customer X has an initial registered user base of 20,000 potential users. This figure is however envisaged to rise to 40,000 users in two years time and potentially to an upper bound of around 60,000 registered users after that. Therefore, the need to architect a platform that can scale to accommodate the growth forecasted for the next two to five years exists.

It is important to understand that the definition of the registered user base does not actually impact the number of users or clients concurrently accessing the solution. Rather, the registered user base is just the user population that may access the solution at any given point in time. Internally, WebSphere Portal Server maintains a database entry for all registered users after their initial login. No constraint, other than the size of the database table and the size of the selected LDAP user repository, should impede the growth of the registered user base. A more meaningful metric when sizing any WebSphere Portal Server solution is the anticipated number of concurrent users or clients. Typically, such values for the number of concurrent clients are calculated as a percentage of the registered user base.

For example, based on the current metrics supplied by Customer X for their existing Web deployment, this figure averages at about 2,500 unique user sessions per hour. This would imply that only 12.5% of the current registered user base actually interacts with the current solution. By the same calculation, the number of concurrent clients would increase to 5,000 for the projected growth in the registered user base to 40,000. This assumes that the percentage of clients using the Portal remains stable at 12.5%. However, careful consideration needs to be taken into account, as this figure may increase once more applications and functions are brought online within the Portal solution. As such, for Customer

X, the actual anticipated estimated rises to 7,500 concurrent clients after two years time, which then increases the percentage to 18.75%.

Normally, it is common for business requirements to state that a Portal should be able to handle X number of clients concurrently.

It is important to distinguish between concurrent clients and concurrent active clients; as such, terminology is often misinterpreted between different parties. Concurrent active clients have both an active connection to the HTTP server as well as at least one thread of execution running in the application server. At any point in time, many of the clients connected to the Portal are not active; they may be thinking, reading, or even drinking coffee. These are considered as inactive concurrent clients, or more generically as concurrent clients. Based on our experience, a good starting point is to assume that for every single concurrent active client there are approximately 10 concurrent inactive clients (1:10). Theoretically, therefore, an application server capable of supporting 100 active clients will support approximately 1000 concurrent clients (active + inactive).

This assumption breaks down somewhat when the characteristics of WebSphere Portal Server shift away from that of being a traditional Web-based solution. For example, a Portal performing more back-end work will effectively shift the assumed work pattern from that of being a traditional Web-based solution to that of an On-Line Transaction Processing (OLTP) solution. Such an OLTP solution will place greater demands on system resources, with a reduced supporting ratio of approximately 1:5 or less.

A further point of debate between different parties is the understanding of Peak Load or Arrival Rate. It is important to recognize that it may be necessary to plan for such situations when many users simultaneously access the Portal solution at the same time. This generally breaks any rule of thumb for concurrency and is indicative of such situations as logins, each morning between 8am and 9am, or campaign launches. Under such circumstances, is it only possible to honor each request by *Planning for the Peak*.

Attention: A sizing estimate should only ever be used as an approximation of the hardware resources required to support the proposed implementation. Actual experiences may vary from the sizing estimate for many reasons. The degree of variability can range from small to very significant. As such, there is no substitute for not undertaking a full capacity planning and performance tuning exercise. Failing to implement this critical part of any project plan is planning to fail.

2.2 The building blocks of an architecture

When faced with the challenge of architecting a WebSphere Portal Server implementation, it is often useful to take a high-level approach to first define the logical components that comprise the very architecture that is about to be designed.

For the experienced IT Architect and Portal Practitioner, this commonly embraces two aspects of design; the component model and the operational model. Component models are typically focused on identifying the components, their responsibilities, and characteristics required to deliver the solution requirements. At a conceptual level, the component model documents the technical architecture at a very high level and does so in a technology agnostic manner. At a specification level, the component model documents the required specifications and corresponding realization of all components, which ultimately will be placed on the operational model, together with a description of their interfaces, dependencies and collaborations. In common terminology, the component model addresses the logical aspects of a solution architecture. By contrast, the operational model provides the description and configuration of the hardware and software technologies needed to deliver the required solution characteristics and capabilities, within the constraints of technology, skills, and budget. It describes the distribution of the solution components onto geographically distributed nodes, together with the connections necessary to achieve the solution functional and non-functional requirements.

Typically, the development of both the component and operational models follow various recognized paths using standard techniques or approaches. However, with the advent of Commercially-Off-The-Shelf (COTS) packages, such as WebSphere Portal Server, the demands on the IT Architect and Portal Practitioner have been reduced. Nevertheless, our experience tells that making mistakes during the architectural phase of an implemention can lead to major consequences later on in a project. As such, it is strongly recommended that IBM is engaged during this crucial period of any implementation, if not at any other time during a project.

2.2.1 Logical Deployment Units

The following Deployment Units are considered in regards to a WebSphere Portal Server architecture. The list, however, is by no means exhaustive and provides only a starting point in recognizing the primary Commercially-Off-The-Shelf (COTS) packages associated with such an architecture.

Internet Browser

The Internet Browser component is a standard Web browser, such as Internet Explorer® or Mozilla Firefox. This component communicates with the solution through the HTTP / HTTPS protocol, receives responses in HTML format, and renders them for the user. The Internet Browser has general characteristics that include Graphical Presentation, HTML, Applet Execution within a Java Virtual Machine (JVM™), JavaScript™ Execution, Plug-In Support, Caching, Security and encryption Services, and Content Persistence (cookies).

Tivoli WebSEAL *(Optional)*

Tivoli WebSEAL is a high-performance, multi-threaded Web Proxy server that applies fine-grained security policy to the Tivoli Access Manager protected Web object space. WebSEAL can provide single sign-on solutions and incorporate back-end Web application server resources into its security policy. WebSEAL normally acts as a reverse Web proxy by receiving HTTP/HTTPS requests from a Web browser and delivering content from its own Web server or from junctioned back-end Web application servers. Requests passing through WebSEAL are evaluated by the Tivoli Access Manager authorization service to determine whether the user is authorized to access the requested resource.

Tivoli Access Manager Policy Server *(Optional)*

The Tivoli Access Manager Policy Server for e-business is an authorization and management solution that scales across the entire enterprise. A robust and secure policy management tool for e-business and distributed applications, it addresses the challenges of escalating security costs, growing complexity, and the need for uniform security policies across platforms. Tivoli Access Manager unites core security technologies around common security policies to help reduce implementation time and management complexity, thereby lowering the total cost of security-enhanced computing.

HTTP Server

The HTTP Server provides the front end to the solution. It allows for greater concurrency and resource off loading from the Portal Server tier, by serving static content (HTML pages, for example) and dynamic content (JSP™ fragments) by way of WebSphere plug-in caching capability. With the Network Deployment configuration, the plug-in provides load balancing among Portal Server cluster members.

WebSphere Application Server

WebSphere Application Server is a Web application server that provides J2EE services for the WebSphere Portal environment. It executes the Java portlets, JavaBeans™, Java Server Pages (JSP) files, and Enterprise JavaBeans™ (EJBs) that are used by WebSphere Portal. In conjunction with two other products in the WebSphere Application Server family of interoperable products (WebSphere Business Integration Server Foundation and WebSphere Application Server Network Deployment), this component is the platform on which WebSphere Portal runs.

WebSphere Portal Server

WebSphere Portal Server is a J2EE application that runs on WebSphere Application Server. Its main function is to serve the WebSphere Portal Server framework to the desktops and mobile devices of portal users. WebSphere Portal Server creates an environment that provides the connectivity, administration, and presentation services that are required. WebSphere Portal Server V6.0.1 includes several new functions and enhancements that make it easier to design, administer, and use.

Web Content Management Server *(Subcomponent)*

The Web Content Management subcomponent of WebSphere Portal Server empowers a knowledgeable workforce by providing an environment that allows them to create, edit, and publish Web content. Because knowledge owners have less dependence on technical resources, they can publish content in a more timely and efficient way by using the Web Content Management component. It is often helpful to think of a WCM server as a stand-alone component due to performance issues. However, licensing restrictions should be checked.

WebSphere Member Manager *(Subcomponent)*

WebSphere Member Manager is the subcomponent of WebSphere Portal Server responsible for accessing the user registries for user and group management and authentication. The user registries may be LDAP servers, a Custom User Registry, or the Member Manager database user registry.

WebSphere Process Server *(Optional)*

WebSphere Process Server (WPS) is a business process integration server that is built on top of the WebSphere Application Server. It is built to support solutions created based on service-oriented architecture (SOA). WPS provides services that enable traditional business integration such as enterprise application integration; it also provides services that enable business process automation, such as choreographing business processes as well as human workflows, and management of those business processes. WPS uses the Service Component Architecture programming model and Service Data Object (SDO) data model. SDO business objects can be defined, transformed, routed, and mapped using SCA components. The connectivity to back-end Enterprise Information Systems (EIS) is provided by the resource adapters. In the outbound mode, WPS uses adapter to send data to the EIS system from the integration application. In inbound mode, WPS uses adapters to trigger the integration application by the event occurring in the EIS system. For example, an adapter can be deployed on WPS to synchronize product information across multiple enterprise information systems. A modification of the product information on one EIS triggers a business application that processes the data and propagates it to the other enterprise information systems.

LDAP Directory Server

A directory is often described as a database, but it is in fact a specialized database that has unique characteristics that set it apart from that of general purpose relational databases. One special characteristic of directories is that they are accessed (read or searched) much more often than they are updated (written). Hundreds of people might look up an individual's phone number, or thousands of print clients might look up the characteristics of a particular printer, but the phone number or printer characteristics rarely ever change.

Database Server

The Database Server's function is to provide persistent data storage and retrieval in support of the user-to-business transactional interaction. The data stored is relevant to the specific business interaction, for example, bank balance, insurance information, current purchase by the user, and so on.

Portlet applications

Portlets are a central part of WebSphere Portal Server. Portlets are small Portal applications that are independently developed, deployed, managed, and displayed. Portlets have multiple states and view modes, plus event and messaging capabilities. Portlets run inside the Portlet container of WebSphere Portal Server, similar to the way a servlet runs on a WebSphere Application Server. The Portlet container provides a runtime environment where Portlets are instantiated, used, and finally destroyed. Portlets rely on the WebSphere Portal Server infrastructure to access user profile information, participate in window and action events, communicate with other Portlets, access remote content, look up credentials, and store persistent data.

J2EE Enterprise Applications

An Enterprise Application is a J2EE deployment unit that bundles together Web Applications, EJBs, and Resource Adaptors into a single deployable unit.

2.2.2 Node characterization at the specification level

It is strongly advised that the specification level attributes for each node in a contending WebSphere Portal Server architecture are clearly defined and documented. As such, each node should be described in terms of the functional and non-functional requirements and how those requirements are met. Table 2-3 gives an overview of node specifications.

Table 2-3 Node specification

Specification Level Node	Example: System x™
Attributes	
Presentation Deployment Units	This section identifies and describes the major DUs (Deployment Units) associated with a node. A DU is considered as a single unit for placement considerations. Furthermore, it is often important to distinguish between the placement of presentation, execution, and data DUs. Example: J2EE artifacts, such as .ear files and .jar files, may be considered as Data DUs, while WebSphere Application Server remains an Execution DU. It is important to recognize that multiple DUs may be grouped together on the same node, where practical.
Execution Deployment Units	
Data Deployment Units	
Environments	Example: Production - Based on 100% of the required NFR capacity.
Hardware	Example: pSeries®.
Operating System	Example: AIX® 5L™ V5.3.0.0-0.3.
Non-Functional Requirements	
Availability	Example: Minimum of two physical nodes, one in each data center, configured as a single active-active cluster across both data centers.
Capacity	Example: Each node should be able to handle 50% of the required capacity. However, as this component is part of the shared common network infrastructure core, the total consolidated capacity must be capable of delivering a guaranteed Quality of Service.
Scalability	Example: Implied horizontal scalability through the addition of extra physical nodes in each data center. Implied vertical scalability for Java based components, hardware resources permitting.
Disaster Recovery and Resilience	Example: In the case of the failure of one physical node, the others will continue to function with a reduction in total capacity. In the case of the failure of a software component on one of the physical nodes, the other collocated software components will continue to function. Depending on the type of failure the recovery characteristics will be different. For example, the failover from a network connection failure has different fail-over characteristics from that of a WebSphere Portal Server cluster member JVM crash.
System Management	Example: Integration of system event monitoring with client X’s enterprise monitoring infrastructure.

2.3 Operational architectures

Increasingly, WebSphere Portal Server customers are interested in deploying a Portal in a business critical environment. However, such a requirement raises the question about how best to address such needs in terms of selecting the most appropriate operational architecture. Fundamentally, these are all aspects that should be defined under the non-functional requirements of a solution. For example, availability requirements should be captured and agreed upon, as early on in a project as possible, as they dictate the high-availability and recovery aspects that an architecture must meet.

The purpose of this section, therefore, is to take a look at how a business critical deployment can be accomplished using today's WebSphere Portal Server V6.0.1 product, and the advantages and disadvantages of each architectural design. It should be noted that regardless of the operational architecture chosen, that there are also a number of high-availability considerations regarding the associated database backup/replication, maintenance procedures, etc. which must be considered in developing a complete operational solution for a WebSphere Portal Server deployment.

2.3.1 Adopting a tiered architecture

A common architectural principle is that of adopting a tiered or segregrated topology. This well practiced approach is in keeping with the J2EE mandate that prescribes the separation of applications into client, presentation, business, and enterprise system tiers. The approach is, however, most beneficial in terms of overall enterprise security and performance optimization.

As such, it is strongly suggested that a n-tier approach is adopted as the topology of choice for all high-volume WebSphere Portal Server deployments. This is regardless of the selected platform. Differentiating between the functional components of the solution allows each physical server to be specifically sized to the task in hand. For example, placing the Web server on a separate physical machine from the WebSphere Portal Server allows each machine to run with different OS characteristics. The same holds true for other server types, such as database servers.

2.3.2 Addressing scaleability and high availability

A major concern with any architecture is the ability to address the needs of scalability and redundancy. Furthermore, it is important to recognize that the operational aspects of just such an architecture, such as availability, also influence the overall design of a solution.

The ability to scale WebSphere Portal Server V6.0.x, or any other WebSphere Application Server for that matter, is essentially achieved by clustering. Clustering allows requests to be Workload Managed (WLM'ed) between a number of cloned copies of the concerned application. In addition, when architected correctly, clustering addresses redundancy and fault tolerance.

The most important factors of a mission-critical production environment are redundancy and fault tolerance, ensuring that there is no single point of failure in an architecture. The most important aspect of fault tolerance is to have at least two of members or replicas of each component. These can either be in a primary-to-backup formation or a peer-to-peer configuration. This thought can even be extended to the data center itself.

There are several approaches for clustering a WebSphere Portal Server Version 6.0.1 implementation. The following section outlines each in detail.

The single clustered architecture

In a standard WebSphere Portal Server V6.0.x clustered architecture, two or more separate WebSphere Portal Server nodes are clustered together to form a single WebSphere Portal Server instance. In turn, each node is capable of supporting multiple vertical cluster members to better leverage the available system resources and to achieve the demands of scaleability. High Availability is thus accomplished not only through the vertical clustering of WebSphere Portal Server, but also by way of horizontal clustering of WebSphere Portal Server to safeguard against the outage of an actual physical node, and the replication of the database and LDAP directory servers, respectively.

As such an architecture utilizes the same user customization, community, and release data throughout an environment, any user customization made against one Portal cluster member by a user would then be available to the same user, as and when that user accesses any of the other cluster members participating in the same Portal cluster. It is acknowledged that under normal conditions, session affinity is maintained against the same Portal cluster member until such time that a user terminates his or her session, or the Portal cluster member becomes unavailable, either through a deliberate or an unscheduled outage.

In isolation, this architecture should be considered the *de facto* WebSphere Portal Server V6.0.x architecture of choice.

However, maintaining continuous operation during periods of scheduled or unscheduled maintenance requires careful consideration. As this implementation does not typically include any redundant hardware, either in the form of a fully redundant production environment or a "double duty" staging environment, maintenance requiring an uninterrupted level of service (also referred to as 24x7 availability) must be performed as a multi-step process.

As such, this involves disabling the automatic file synchronization service from the WebSphere Deployment Manager administrative admin console and then stopping the node agent on each of the nodes participating in the cluster. Maintenance is then performed on each node in turn, starting with the primary node, by first gracefully quiescing user requests from each node by modifying the WebSphere Web server plug-in load balancing weighting (when multiple cluster members exist on the same node, all must be stopped at the same time) while the remaining node or nodes in the cluster continue to honor user requests. The final step is to synchronize and restart all of the nodes one at a time, not forgetting to re-enable the automatic file synchronization service.

While this approach represents a distinct improvement over the 24x7 maintenance procedures applicable to previous versions of WebSphere Portal Server, the complexities of performing maintenance and maintaining an uninterrupted level of service arguably remain high risk for many organizations. As such, the decision to implement this approach rests with the comfort factor of each particular organization.

Figure 2-2 on page 23 illustrates the system topology needed for a WebSphere Portal Server V6.0.x single clustered architecture.

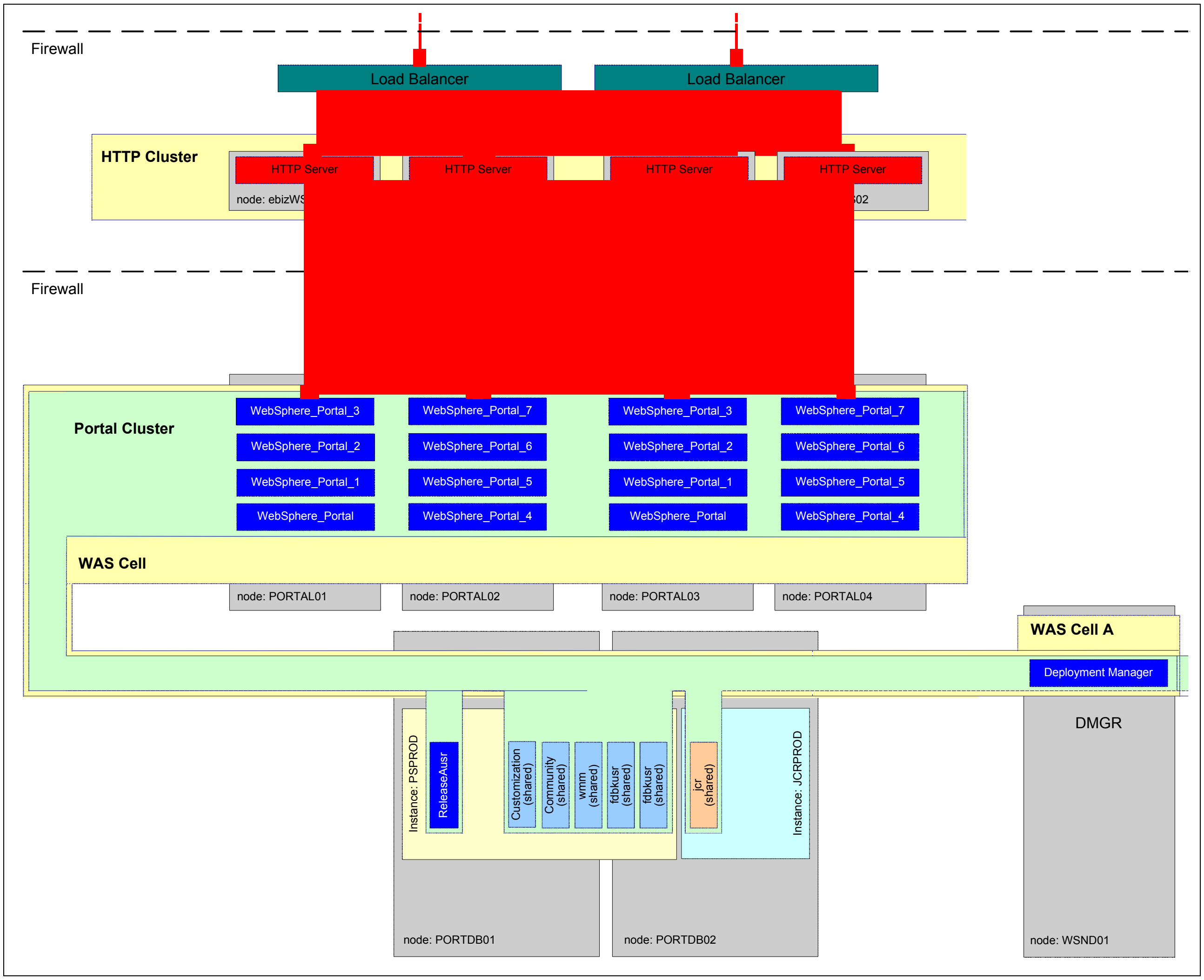


Figure 2-2 A single clustered architecture

Key features of this architecture are:

- A single load balanced HTTP Server cluster (HTTP Cluster) that spans two or more physical nodes.
- A single WebShere Portal Server cluster (Portal Cluster) deployed in a single WebSphere Cell.
- The WebShere Portal Server cluster consists of two of more horizontal cluster members and any number of vertical cluster members per node (resources permitting).
- A dedicated stand-alone WebSphere Deployment Manager is responsible for the management of the entire WebSphere Cell (Cell A).
- As the environment only consists of a single WebShere Portal Server cluster, only a single release database domain is required.
- The remaining database domains (communityusr, customizationusr, wmmusr, fdbkusr, lmdbusr, and jcr) are deployed alongside the release database domain. Note that the JCR Repository exists in a different database.
- The environment also hosts a LDAP directory server (not shown), which is highly available, for maintaining the registered user base.

The multiple clustered architecture

New to WebSphere Portal Server Version 6.0.1 is the ability to architect multiple Portal clusters within the same WebSphere cell. Indeed, the WebSphere Portal Server Version 6.0 Information Center describes just such an architecture and the necessary configuration tasks needed to implement such a deployment.

It is, however, important to understand that such an architecture is subject to a number of inherent limitations. Most important, despite the fact that the Information Center states that it is possible to federate multiple, independently configured Portals into the same WebSphere cell and that it is possible to manage such clusters from the same cell, it must be recognized that only a single J2EE enterprise application, of a unique name, can be deployed into a given WebSphere cell at any one time. Furthermore, all J2EE enterprise applications are cell-scoped. This limitation makes it impossible to deploy different versions of the same enterprise application against the different Portal clusters within the same cell, as the case might be during periods of 24x7 maintenance. This extends to WebSphere Portal Server itself, which consists of a number of enterprise applications that make up the effective runtime and also to the very Portlet applications deployed within the solution.

In other words, enterprise applications are shared across the WebSphere cell, regardless of the presence of multiple Portal clusters or not. As a consequence, it is not possible to upgrade one Portal cluster in isolation from another, as the underlying enterprise applications and supporting class libraries are common to both. Attempting to do so runs the risk that incompatibilities will result, potentially bringing down the complete environment.

It is plausible that such a WebSphere Portal Server architecture is practical for the purposes of providing different applications and services between Portal clusters, which might be the case with each Portal cluster supporting a different line of business.

The dual cluster with two lines of production architecture

Deploying either a single clustered instance or a multiple clustered instance within the same WebSphere Cell of WebSphere Portal Server V6.0.x would at first glance appear to be the most logical architectural choice for most implementations.

However, when 24x7 availability is considered as a non-functional requirement, both architectures fall short of the mark. This is not to say that both architectures are not capable of maintaining a continuous level of operation during periods of either scheduled or unscheduled maintenance. Indeed, WebSphere Portal Server V6.0.x has introduced considerable improvements over previous versions in this very respect. Rather, the considerations are associated with the comfort factor demanded by the very organizations that typically manage such implementations. For those organizations that do not mandate 24x7 availability, which should not be confused with high availability, deploying a single clustered instance of WebSphere Portal Server remains an acceptable choice.

The deployment, therefore, of a WebSphere Portal Server V6.0.x dual clustered architecture represents a significant architectural enhancement for the majority of organizations looking to meet the needs of continuous operation with a 24x7 availability requirement. Indeed, such an architecture represents the new WebSphere Portal Server V6.0.x Gold Availability Standard. The primary enabling feature of this capability is the presence of two distinct WebSphere Portal Server clusters, each deployed within separate WebSphere cells. This makes it possible to perform maintenance on one Portal cluster, while the other continues to service user requests. The benefits of adopting such an architecture are that there are effectively two “Lines of Production”. This allows for continuous operation during periods of maintenance, albeit at reduced capacity, without the need for a dedicated failover environment. One “Line of Production” can effectively be taken off line, as and when required, without impacting the remaining “Line of Production”. Critically, the ability to share user customization and community data between different Portal clusters, and the cluster members that participate in each, ensures consistency at the user interaction level (new to Portal Server V6.0.x). That is, any user customization made against one Portal cluster member, by a user, is now available to the same user, as and when that user accesses any of the other cluster members participating in the same or different Portal cluster. Each Portal cluster, in turn, maintains a separate release repository, which provides a mechanism for maintaining all Portal resource definitions, rules, and rights specific to that cluster.

Again, this is an acknowledged product improvement from the limitations associated with WebSphere Portal Server V5.1.x and earlier. Failing to implement such an architecture would otherwise necessitate deploying a single clustered instance of WebSphere Portal Server and adopting either the IBM documented 24x7 Portal maintenance procedure or the use of a secondary maintenance environment.

Figure 2-3 illustrates the system topology needed for a Portal Server V6.0.x dual cluster architecture.

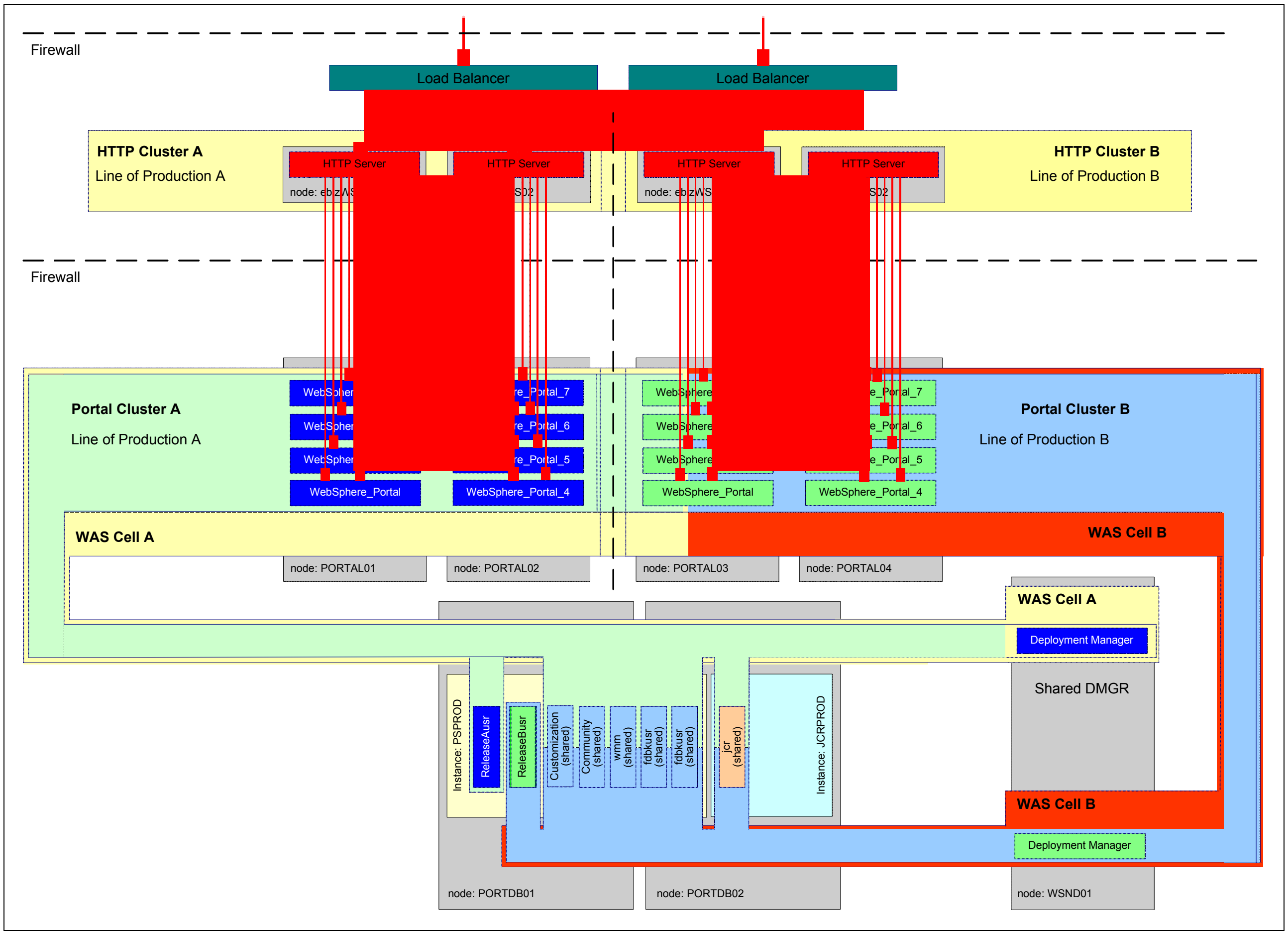


Figure 2-3 Dual cluster architecture illustrating two lines of production

Key features of this architecture are:

- Two independent HTTP Server clusters (HTTP Cluster A and HTTP Cluster B), consisting of at least two physical nodes per cluster (so that each cluster is highly available in its own right).
- Two independent WebSphere Portal Server clusters (Portal Cluster A and Portal Cluster B), one per WebSphere Cell.

- Two independent WebSphere Cells (Cell A and Cell B).
- Each WebSphere Portal Server cluster consists of at least two physical nodes per cluster or cell (so that each cluster is in highly available its own right).
- The WebSphere Plug-in resident in each HTTP Server only routes requests to the cluster members for the immediate Portal Cluster.
- Two independent WebSphere Network Deployment Manager (Deployment Manager) instances, one per WebSphere Cell, are collocated on the same physical node.
- A separate release database domain (releaseAusr and releaseBusr) exists for each WebSphere Portal Server cluster or “Lines of Production” (Portal Cluster A and Portal Cluster B), maintaining indenpendant configuration data for each.
- The remaining database domains (communityusr, customizationusr, wmmusr, fdbkusr, lmdbusr, and jcr) are shared between each WebSphere Portal Server cluster or “Lines of Production” to maintain a consistent user experience. Note that the JCR Repository exists in a different database.
- The environment also hosts a LDAP directory server (not shown), which is highly available, for maintaining the registered user base.

It is worth noting that a dual clustered architecture will require twice as much administration as a single clustered deployment. Furthermore, in order to keep each “Line of Production” in synchronization, a staging environment plays an important part for preparing build promotions. Such tools as XMLAccess and Release Builder must be utilized to ensure consistency between the different “Lines of Production” or clusters.

The geographically deployed architecture

As WebSphere Portal Server has evolved, one requirement that has continually been requested has been the ability to deploy an architecture in geographically distributed fashion. With the release of WebSphere Portal Server V6.0.x, this requirement is now a possibility.

Such a requirement, however, raises the question about how best to design an operational architecture that caters for such a "global deployment".

Not every WebSphere Portal Server deployment with a geographically scattered workforce requires that the physical servers themselves are geographically dispersed. Indeed, many internet facing Portals may be country or region specific, but by the very nature of the internet are accessible worldwide. However, when the demands of an implementation start to include the very integration points that a Portal brings together, partitioning across geographies becomes a necessity. Aspects, such as high availability and disaster recovery, are also influencers for considering a distributed implementation. For example, partitioning a WebSphere Portal Server deployment between Europe and North America becomes a prerequisite when each major geography maintains the local services and back-end systems that are effectively accessed through the Portal. The idea of a geographically deployed architecture can also be considered in terms of a split between multiple data centers, even when those data centers exist within close proximity to one another (across the street or across a city).

With the introduction of database domains in WebSphere Portal Server V6.0.x, greater flexibility was made possible in terms of the permissible operational architecture. As such, the distinction between release, community, and user customization data has made it possible to achieve a truly "global deployment". Readers familiar with previous versions of WebSphere Portal Server will recall that it was not possible to split the Portal database between multiple redundant clusters, located potentially in different geographies, and to maintain a consistent user experience. Indeed, such an architecture when deployed sacrificed the ability for a user to make any customization or personalization modifications, as the changes simply could not be propagated between clusters. This in part was attributed to the fact that the internal object IDs associated with the various elements of a deployment could not be guaranteed to be unique. Any attempt, therefore, to deploy a bi-directional database replication technique was further hindered. To overcome this constraint, it was mandatory that all Portal cluster members, participating in the same Portal instance, accessed the same centralized database. However, the performance considerations of accessing a centralized database across the WAN from geographically dispersed Portal servers made this approach impractical in many environments.

In addition to the separation of Portal data into distinct database domains with WebSphere Portal Server V6.0.x, which represents an acknowledged product improvement, it should also be recognized that Portal data can now be shared between different Portal clusters and the very cluster members that exist within them. Such database domains can now be deployed in a peer-to-peer manner using techniques like queue replication or 2-way SQL replication in order to provide a global deployment capability, where user personalization is automatically made available to all Portal clusters in all geographies. In this manner, WebSphere Portal Server V6.0.x also allows users to experience portability should they temporarily access the Portal solution from another geo (branch or office location).

> **Important:** The implementation of a multi-clustered WebSphere Portal Server V6.0.x architecture that sees the deployment of individual clusters in each geography to support a truly "global deployment" mandates the use of the same LDAP directory server in all geographies. The LDAP directory server may, however, be replicated for redundancy purposes. This requirement is necessary to maintain uniqueness between Portal users.

The WebSphere XD deployment architecture

The latest WebSphere Portal Server V6.0.1 deployment option includes support for WebSphere Extended Deployment V6.0.2, or WebSphere XD for short. Such an architecture makes it possible to dynamically start and stop additional WebSphere Portal Server cluster members, or dynamic clusters in the WebSphere XD sense, as and when workload demands. In addition, the On-demand Router (ODR) component of WebSphere XD can be utilized to route requests based on priority and user rules.

For more information about WebSphere Extended Deployment refer to:

- WebSphere Extended Deployment (XD) 6.0.2 support for WebSphere Portal Server, found at:

  http://www.ibm.com/support/docview.wss?uid=swg21264596

- WebSphere Extended Deployment (XD) 6.0.x Information Center, found at:

  http://www.ibm.com/software/webservers/appserv/extend/library/library60x.html

2.4 Portal deployment considerations

Three principle methods exist for implementing maintenance in a WebSphere Portal Server V6.0.x production environment.

2.4.1 In-situ maintenance procedures

As the name suggests, in-situ maintenance describes the process of undertaking maintenance in a target environment without the inclusion of an additional environment for providing operational continuity. For the most part, such maintenance may simply be undertaken during a period of scheduled outage, such as during the weekend or overnight, when the respective users of the solution may be unaffected by any downtime. Alternatively, in-situ maintenance can be performed by adhering to the IBM documented 24x7 maintenance procedure. However, while this latter approach represents a distinct improvement over the 24x7 maintenance procedures applicable to previous versions of WebSphere Portal Server, the complexities of performing such maintenance, and maintaining an uninterrupted level of service, arguably remain high risk for many organizations. As such, the decision to implement the 24x7 maintenance procedure in a single clustered WebSphere Portal Server V6.0.x environment can only rest with the comfort factor required by each particular organization.

The key differentiator between this and the other maintenance methods described in this section is that the in-situ procedure does not require an additional environment during the period of maintenance. Full details for this procedure, including the IBM documented 24x7 maintenance proceedure, can be found in the WebSphere Portal Server Version 6.0 Information Center.

2.4.2 Two sets of production environments

This option considers two sets of production environments, each supporting duplicate WebSphere Portal Server configurations. It is not implied that either environment shares the same user community, customization, and wmm database domains in a peer-to-peer fashion, as is now achieveable with WebSphere Portal Server V6.0.x. Instead, it is characterized by the replication of all data and artifacts between hosting environments prior to undertaking maintenance.

As such, the deployment of two sets of production environments has long been an acknowledged approach for maintaining operational continuity and for addressing disaster recovery. However, unlike the approach detailed in 2.4.3, “The dual cluster with two lines of production architecture” on page 29, such a deployment does not normally see both sets of production servers actively handling the load. It is what is commonly referred to as an active/passive implementation. Furthermore, the passive environment may also serve the purpose of addressing disaster recovery.

When considering an architecture based on the selection of two sets of production environments, there are two sub approaches that are most commonly implemented.

The Flip-Flop approach

One approach for maintaining a continuous level of operation in a WebSphere Portal Server environment has been with the adoption of a Flip-Flop architecture. Such an architecture utilizes a second WebSphere Portal Server instance identical to the primary instance. Each environment is normally clustered and highly available in its own right. During scheduled maintenance, user requests are first gracefully quiesced over to the secondary instance, or Flip environment, before Fix Packs and updates are applied to primary instance. With the successful completion of all maintenance activities to the primary instance, the decision can be made to Flop the users back or to retain the users against the current secondary instance. If the latter option is selected, the secondary WebSphere Portal Server environment effectively becomes the primary instance.

Unfortunately, there are several weaknesses associated with the Flip-Flop architecture that make the approach somewhat less than straight forward to implement. First, it should be fairly evident that the procedure requires an additional physical environment. Typically, many organizations address this issue by collocating the secondary instance, or Flip environment, on the same hardware hosting their staging environment. Secondly, there is the requirement to transfer all configurational data, including the very Portlets deployed within the Portal, and the need to carry across any user customization data between Portal instances. With the caveat that WebSphere Portal Server prior to V6.0.x did not support database domains, the possibility that such data could be readily shared between Portal instances was not feasible; the only option was the one-way transfer of such data between environments. Finally, the need to undertake any so-called backend plumping, to those systems and services being integration through Portal, warranted significant time and effort to ensure a satisfactory outcome.

Environments with multiple personalities

This implementation deploys both a production and a staging environment. The staging environment, however, has multiple personalities and pulls "double duty". It is primarily the staging environment but also acts as a standby pseudo-production environment for times of scheduled maintenance to the production environment. There is obviously some planning required to ensure that the environment is up to date with configurational data from the production environment, prior to being put into service. The same limitations regarding the replication of data between environments, as described under the Flip-Flop approach, also hold true. Furthermore, this approach is really only viable if the staging environment mimics the production environment in terms of overall system resources and capacity.

It is also vitally important to recognize that it is not the actual staging instance of WebSphere Portal Server, as such, that handles the temporary production load. Rather, such an implementation necessitates the installation of a secondary instance of WebSphere Portal Server alongside that of the staging instance. Without such a configuration, production and staging data would merge and impact the underlying integrity of the entire solution.

2.4.3 The dual cluster with two lines of production architecture

The deployment of a dual clustered WebSphere Portal Server V6.0.x architecture with “Two Lines of Production” brings about distinct advantages when maintenance and operational continuity are concerned. Indeed, the architecture sets the new Gold Standard for Availability with WebSphere Portal Server V6.0.x. The approach makes full use of the WebSphere Portal Server V6.0.x product enhancements that introduce the concept of database domains. As such, one “Line of Production” can be effectively taken off line, as and when required, without impacting the remaining “Line of Production”. Furthermore, as each “Line of Production” has its own release database domain, while the user community and customization database domains are shared, this makes it possible to have two different releases in production.

The deployment of a dual clustered WebSphere Portal Server V6.0.x based architecture usually consists of a minimum of four physical nodes. The nodes are split into two halves (each half will consist of two physical nodes and will host what will effectively be a separate but identical Portal cluster). Customization and community data is shared between each peer cluster permitting user customization updates made in one cluster to be available to the peer. However, release data is maintained on a per cluster basis. Further details can be found in “The dual cluster with two lines of production architecture” on page 24.

Attention: It is important to recognize that the deployment of a dual clustered WebSphere Portal Server V6.0.x architecture with “Two Lines of Production” introduces special considerations in terms of code deployment and release management. Such a requirement is needed to ensure that each “Line of Production” remains consistent and identical in terms of overall user experience. This is achieved by assembling a build, or release, first in a staging environment and then by promoting the build, or release, to each Line of Production.

2.4.4 Moving a configuration between environments

A common deployment approach in any IT implementation is to provide separate environments for development, quality assurance, performance testing, pre-production, and production (or some subset of these). As applications move through this life cycle, there is the need to repeatedly promote code and configuration data between environments. Furthermore, the need becomes even more important with the exploitation of a dual clustered WebSphere Portal Server V6.0.x architecture with “Two Lines of Production”.

As such, there are several methods for replicating WebSphere Portal Server configuration data between environments. However, it is important to understand the limitations and assumptions associated with each method.

To get an exact replica of one environment to another, it is first necessary to make a full XMLAccess export from the source environment. At this point, you could import the XML file resulting from the previous XMLAccess export action into another Portal instance to create what would at first seem to be a duplicate configuration. However, as the import action by default applies the literal object IDs associated with the resources from the source environment, there is a high likelihood that these object IDs will clash with any existing object IDs in the target environment. You could avoid this by using the ID generating mode (see the XML reference documentation for detail). When you use the ID generating mode, the object IDs in the input are not taken literally; instead, during the import process, the resources obtain new object IDs when they are created on the target system. However, while this prevents any clash in terms of the object IDs, it does not yield an exact replica. In certain situations, this may be sufficient for a number of customers.

The recommended approach, therefore, for creating an exact replica of one environment to another involves the additional task of “cleaning” the target environment. As such, the target environment needs to be an empty Portal void of any object IDs. Importantly, this must only be undertaken on the target environment after it has been configured completely (including activating security, WCM, and so on). This then allows the literal object IDs from the source environment to be restored as is. Using literal object IDs only makes sense if you really want to create two instances of the same resource, and if you have a controlled environment where you can guarantee that all object IDs that your resources depend on have exactly the required values.

In addition to any XMLAccess and Release Builder tasks, it is necessary to manually copy the associated Portal artifacts between environments. For example, it is necessary to copy the Portlet WAR (Web Archive) to the installableApps directory of the target environment.

2.5 Architecting for performance

WebSphere Portal Server architectures that perform and scale are not accidental. They are the result of proper design development and rigorous assurance processes that include iterative performance verification and re-verification during the entire solution life cycle.

The importance of addressing performance as early in the project as possible is crucial to project success. A positive result requires a strong effort to ensure that performance maintains visibility and importance throughout the project life cycle. Performance, like security, is a secondary consideration on too many projects, resulting in failure, lower capabilities, or frequent critical situations (that IBM must often help to resolve). Such results need not occur if performance is correctly handled and supported from the outset of the project.

2.5.1 Scalability

As mentioned previously, the ability to scale WebSphere Portal Server V6.0.1, or any other WebSphere Application Server for that matter, is essentially achieved by clustering. Clustering allows requests to be Workload Managed (WLM'ed) between a number of cloned copies of the concerned application. In addition, when architected correctly, clustering addresses redundancy and fault tolerance.

Overall Portal scalability will be accomplished by clustering at the various different tiers of the architecture, both horizontally and vertically where permitted. Usually, a platform’s capacity must be capable of handling a realistic amount of transactions over a projected time frame.

2.5.2 Guidance regarding vertical and horizontal scaling

Ultimately, as client load increases, even a properly tuned WebSphere Portal Server will reach a point at which throughput reaches a maximum. After this point, throughput remains fairly constant while response times degrade as further clients attempt to access the solution. Eventually, a saturation point will be reached. The number of concurrent active clients at the saturation point represents the maximum active client concurrency for the solution. Adding more nodes into a cluster is one way of scaling an application to achieve the goals defined by the non-functional requirements (NFRs).

Vertical clustering

Vertical clustering should be considered for a number of reasons:

- To fully utilize the processing power of modern SMP servers
- Local redundancy

Horizontal clustering

By contrast, horizontal clustering should be considered for the following reasons:

- To achieve scalability beyond the limitation of individual servers
- Redundancy and reliability
- Hardware failover

Horizontal scaling is especially effective in environments that contain many smaller, less powerful machines. Client requests that would overwhelm a single small machine can be distributed over several machines in the solution. Failover is another benefit of horizontal scaling. If a machine becomes unavailable, its work can be routed to other machines containing cluster members.

By contrast, vertical cloning benefits Symmetric Multi-Processing (SMP) systems and should be implemented when system resources are found to be underutilized. For Java, systems with multiple processors typically outperform multiple systems with fewer processors, when comparing the total number of processors.

A general rule of thumb has always been to allocate a single Java Virtual Machine (JVM) to a single processor. Traditionally, this was based on the constraint that the compaction phase of Garbage Collection (GC) was single threaded. Recent versions of the JVM have, however, minimized this restriction, potentially allowing a greater number of JVMs to run given the total number of available processors. It is also worth remembering that there are a number of JVMs associated with the Deployment Manager and NodeAgent (a NodeAgent is required for each individual physical node participating in the cell) when running Portal Server V6.0.1 in a clustered environment.

Tip: Our experience has shown that many customers fail to implement vertical clustering when horizontal clustering is implemented to address the needs of high availability. As such, it is an IBM recommended best practice that both vertical and horizontal clustering are implemented to address the needs of scalability, high availability, and operational availability.

2.5.3 WebSphere queuing mechanism

In order to understand how to maximize performance, it is necessary to understand the WebSphere queuing mechanism. WebSphere implements a componentized architecture, channeling requests through a number of queues. These queues or pools include a Proxy server and Web server (considered even though they are external components), the WebSphere Application Server embedded Web Container, the EJB™ container, data sources, and possibly other connection pooling mechanisms to various custom back-end systems. Each of these resources sustains a queue of requests waiting to use the resource in question. The overall queuing mechanism is designed to converge towards the back end, where resources are deemed more expensive. For example, out front it is not uncommon for the Web server queue to be configured to handle an inordinately large number of requests. This contrasts to a data source pool, which by nature is more expensive (both in terms of CPU and memory) and thus usually only configured to handle a maximum of 10-20 connections simultaneously. Each queue has the potential to become saturated. There also exists the possibility that if one of the back-end queues saturates, that it will have an effect on the other queues in front. For example, it is not unusual that if a data source connection pool saturates, that the Web container will also eventually overload (simply due to the fact that requests cannot be processed further downstream). This can be particularly confusing when investigating performance. In which case, it is recommended that you take a holistic approach to performance tuning and determine which queue saturates first.

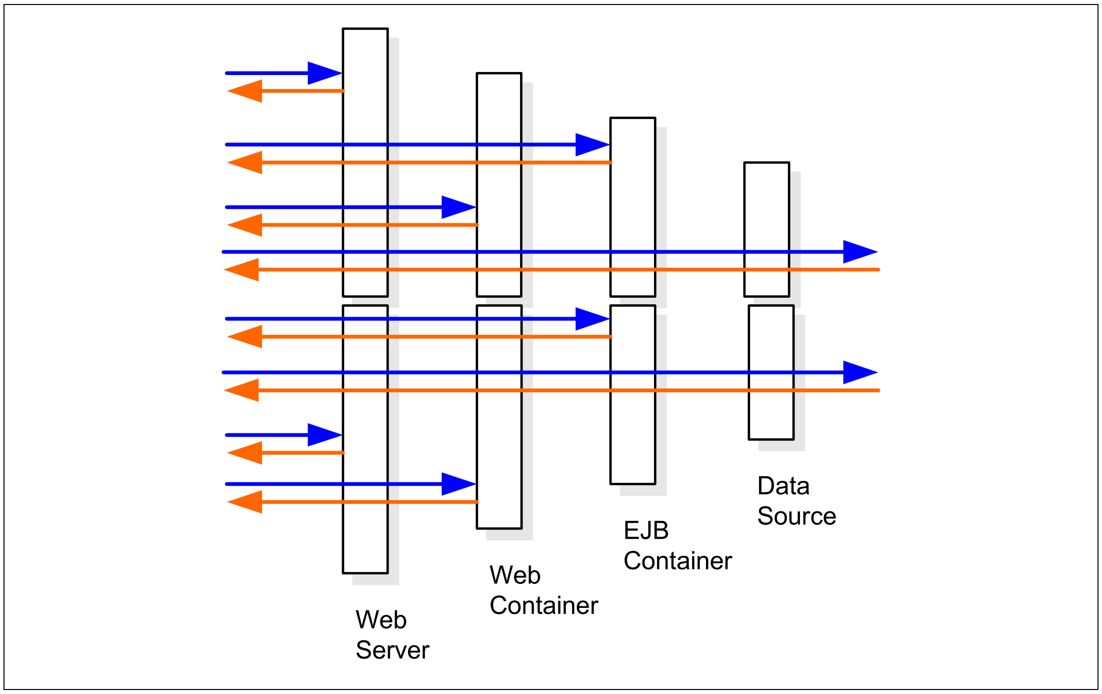


Figure 2-4 WebSphere queuing mechanism

The ability to queue requests in the network layer is a critical part of the WebSphere queuing mechanism. For example, if there are more connection requests than available Web container threads, then connections start to backlog, waiting for threads to be freed. If the maximum number of backlog connections is reached, new connections will be refused. Increasing the MaxConnectBacklog queue can extend the number requests queued in the network layer. However, the full implication of increasing the MaxConnectBacklog queue should be understood, as an increased value can effectively lead to latency problems with the WebSphere plug-in resident in the chosen Web server not immediately detecting that an application server has ceased operation (either through a deliberate stoppage or on the occasion of a JVM crash). Choosing the most appropriate value is therefore dependant on the results you wish to achieve.

2.5.4 Choosing a platform

The selection of a suitable platform has many factors that need to be taken into account. The precedent is, however, usually set by cost and the immediate skill set of the team that will install and administer the solution.

The following points should be carefully considered when making choices between hardware platforms:

- Cross platform benchmark comparisons should be done with caution. Hardware and software differences between platforms may result in inappropriate comparisons.
- If comparisons are made, pay special attention to clock speed, number of CPUs used, and hardware manufacturer benchmarking data.
- Take into account aspects beyond raw clock speed, including threading models, network and disk I/O, instruction cache hit/miss ratios, memory speeds, and so on.
- AIX and Solaris™ platforms both have much better scalability than non-UNIX platforms.
- The Linux® version is a factor. The most recent versions have incorporated scheduler changes that contribute to performance.
- To achieve increased Linux performance, it has been found necessary to update to the latest kernel and to reduce the number of Web Container threads.

Of course, actual performance depends on complex interrelationships among many variables, including the underlying operating system characteristics.

It is strongly suggested that a three-tier approach is adopted for the solution. This is regardless of the selected platform. Differentiating between the functional components of the solution allows each server to be specifically tailored to the task in hand. For example, placing the Web server on a separate physical machine from the application server allows each machine to run with different OS characteristics. The same holds true for other server types, such as the database servers.

It is also worth considering the licensing implications associated with choosing the platform, as WebSphere Portal Server is typically licensed, at the time of writing, by user or by CPU. As such, there is no distinction regarding the actual performance of a CPU.

2.5.5 Separation of WCM from Portal Servers

Although WCM is an integrated sub-component of WebSphere Portal Server V6.0.1, for reasons attributed to performance and scalability, one IBM recommended best practice is that WCM is externalized in its own instance. This approach allows the primary WebSphere Portal Server instance, which maybe clustered, to concentrate on performing the core Portal tasks without WCM resource impact. When WCM is installed in such a manner, you still need to install an underlying WebSphere Portal Server runtime specifically for WCM. However, such a WebSphere Portal Server runtime does not participate in the primary Portal Server instance. Note that this additional WCM Portal Server instance does not participate in the same WebSphere Cell as the primary WebSphere Portal Server instance. As such, if WCM clustering is a requirement, it is also necessary to install an additional instance of WebSphere Network Deployment Manager to overcome this restriction.

Separating the functional aspects of the primary WebSphere Portal Server instance and the WCM instance allows for loose coupling between components. As such, one immediate benefit comes about when upgrading either component; there is no interdependency and each component can be upgraded in isolation. WCM is intended first and foremost as a Web content management system (Web-CMS). As such, it is primarily designed for creating, managing, and publishing Web content consisting of text and images. This should not be confused with the functionality provided by Enterprise Content Management (ECM) solutions.

Deciding to run WCM as an integrated sub-component of Portal Server V6.0.x remains a valid and supported IBM option. However, care should be taken as this approach may place a higher demand on processor utilization and the JVM heap.

Unlike previous versions of WebSphere Portal Server, prior to V6.0.x, which ran WCM as an integrated sub-component, there is no longer the need to create a separate WCM JCR database repository for each Portal Server cluster member. For a cluster consisting of six members there is only the need for a single shared JCR repository. As such, storage capacity requirements are reduced when compared to previous releases.

2.5.6 Separation of Web servers and WebSphere Portal Servers

In most cases, unless the hardware cost is a limiting factor, it is an IBM recommended best practice to architect the Web server and WebSphere Portal Server on separate physical nodes. This allows the greatest level of availability and performance. The Web server nodes may be specifically tuned for static content serving. The major architectural advantage, however, of such a configuration is that the Web server nodes may be placed within a DMZ, and the WebSphere plug-in can communicate through the internal firewall to the WebSphere Portal Server nodes located within another segment of the corporate LAN. Separating the Web server from the WebSphere Portal Server also reduces any contention between resource utilization. Potentially, this would allow a dedicated WebSphere Portal Server to perform without the impact attributed to the co-location of the Web server and vice versa.

However, despite these limitations, co-location remains a valid option in low volume environments within an architecture for intranet applications requiring Web server facilities over and above those provided by the embedded Web container. It is not recommended that the embedded Web container is accessed directly.

Surpassing the saturation imposed by a single Web server is easily and cheaply achieved by architecting additional Web servers. These nodes are typically commodity based machines (in comparison to mid and high-end UNIX® servers). Linux is well suited here, although at a price' AIX also offers the benefit of the Fast Response Cache Accelerator (FRCA) kernel based caching mechanism.

Architecting a minimum of three Web servers is also recommended from the point of view that, if a Web server should fail or be taken out of service in a two-server model, then the remaining server has the potential to become overloaded. Load balancing is most effective with three or more servers in a cluster. As such, applications that require load balancing should ideally be spread across three or more servers.

2.5.7 JVM recommendations

One common misnomer is that setting a large JVM heap size improves performance. This is simply not the case. It is strongly advised that the Java maximum heap setting is chosen carefully and then only based on a thorough Java garbage collection (GC) analysis.

Remember:

- If you use a big heap, then garbage collection will be less frequent but much slower, as there is a lot of memory to search through.
- If you use a small heap, then garbage collection will be more frequent but very fast, as there is less memory to search through.

The Java garbage collection (GC) cycle, which is a “stop-the-world” implementation, will prevent the application server from handling loads for a short period of time. All threads are effectively suspended, with the exception of the garbage collection threads, while GC completes to protect the Java heap from corruption. WebSphere Portal Server vertical clustering can be used to ensure that the CPU is able to provide execution time for at least one cluster member server that can handle the load. Since Version 1.3.x, the IBM JVM has supported multiple garbage collection (GC) helper threads to improve performance during the mark phase of GC.

A major concern for IBM is when customers configured WebSphere Portal Server with a large JVM heap and a high Web Container thread pool. In keeping with the IBM Proven Performance Tuning Methodology, the recommendation is to reduce the JVM heap and the Web Container thread pool and to distribute the load over additional cluster members. The larger the number of Web Container threads, the higher the number of concurrent requests allowed to enter the Web Container. At some point, however, the number of concurrent threads being processed by the Web Container may overwhelm the ability of the JVM. To prevent such an occurrence, a smaller Web Container thread pool can be used. Pending requests will be queued in the network layer.

The rationale behind this recommendation is that smaller discrete cluster members will generally outperform one larger single occurrence, so sharing the load equally guarantees better concurrency. In addition, the benefit from running a single JVM on a large multi-processor machine does not always benefit from all of the resident CPUs. Of course, there are many other factors that influence performance. It should be remembered that adding additional cluster members is the method by which WebSphere and WebSphere Portal Server scales.

It is important to mention that garbage collection (GC) is one of the strengths of Java. By taking the burden of memory management away from the application developer, Java applications tend to be much more robust than applications written in non-garbage collected languages. However, this does not mean that the Java developer can totally neglect memory management. Failing to dereference Java objects after use will prevent the Java garbage collector (GC) from freeing the memory back to the Java heap (this constitutes a memory leak). Likewise, fetching large resultsets and placing them into an array, so that they can be passed as a single variable between objects, also has memory implications.

2.5.8 Portlet application JVM considerations

Portlet applications, like any other Java based applications, when deployed into WebSphere Portal Server, reside within the same JVM and therefore share resources, such as JVM heap space and the Web container thread pool. In this way, Portlet applications are limited to the following constraints:

- Shared JVM resources may become constrained if one Portlet application experiences runaway memory consumption.
- The JVM cannot be tuned specifically for any one Portlet application's requirements.
- Individual Portlet applications cannot be guaranteed a QoS above that of the WebSphere Portal Server proper.
- Poorly written Portlet applications cannot be isolated and potentially run the risk of impacting other Portlet applications deployed within the Portal.
- Of course, Portlet applications that perform and scale are not accidental. They are the result of proper development and processes that include iterative performance verification and re-verification during the entire application life cycle, including post-production maintenance.

2.5.9 High availability and HTTPSession failover

User interactions with WebSphere Portal Server are maintained through the use of a HttpSession. This provides a way to preserve data across multiple pages or requests on an individual user basis. The failure or outage, either scheduled or unscheduled, of a WebSphere Portal Server cluster member will result in the termination of the user's HttpSession. As such, it is possible to enable HttpSession failover support to facilitate maintaining a user's session when requests are failed over to a subsequent cluster member.

However, arguably one of the most misunderstood subjects is that of the HttpSession. First, for WebSphere Portal Server, the HttpSession should not be confused with the LTPA token. It is the actual LTPA token and not the HttpSession that maintains the delegated authentication credential. Without such a credential, a user would be challenged to reauthenticate with the solution each time he or she initiates a subsequent request. The HttpSession is also subject to an inactivity timeout. However, it is possible to configure Portal Server to create a new HttpSession should the initial HttpSession expire. Secondly, the use of the HttpSession as a mechanism within WebSphere Portal Server for persisting user attributes for the duration of a user's session is generally discouraged, as there are alternative methods for achieving the same goal more efficiently.

It should be noted that HttpSession failover does not provide transaction failover. In-flight transactions would need to roll back in the event of a cluster member outage, regardless of whether HttpSession failover was enabled or not. It is, however, worth remembering that the underlying WebSphere Application Server instance of WebSphere Portal Server includes a Transaction Manager that can be leveraged. Finally, it follows that the size of the HttpSession object and the size of the permissible Java heap directly influence the number of users that Portal can concurrently support. Of course, scalability issues can be addressed by WebSphere clustering.

2.6 Security

Security within the enterprise has become increasingly more important and complex as distributed systems and Internet technology have merged. The issue can hardly be ignored, as security breaches are announced in the news on a daily basis. While security is becoming increasingly more complex, technology has also provided us with better ways to implement and maintain security within an organization.

2.6.1 WebSphere Portal Server security

WebSphere Portal Server leverages the underlying security mechanisms of WebSphere Application Server for authenticating users. That is, when a user logs into the Portal, it is actually the underlying WebSphere Application Server that performs the authentication task (assuming that no Reverse Authenticating Proxy Server, such as Tivoli WebSEAL or CA SiteMinder are being used). WebSphere Portal Server then goes on to retrieve the security context from WebSphere Application Server and processes the login. Then, the integrated WebSphere Member Manager component of WebSphere Portal Server must perform an additional LDAP query to retrieve a further number of user attributes and to determine the group membership for the concerned user.

Successfully authenticated users also receive a Lightweight Third-Party Authentication (LTPA) token, containing a delegable credential in the form of an encrypted transient cookie, from the underlying WebSphere Application Server instance. This cookie is only valid for the duration of a user's browser session and is used by way of the embedded LTPA token, to honor subsequent requests which would otherwise require re-authentication. However, the LTPA token is in itself subject to expiry even if a user's browser session is maintained. The LTPA token effectively starts to time out immediately upon creation.

WebSphere Portal Server also includes all the functionality for controlling access to resources based on a number of predefined roles. This involves the process of both determining if the identified requester has permission to access the requested resource and the ability to make fine-grained authorization decisions.

2.6.2 Using External Security Managers

Although not a mandatory requirement for a WebSphere Portal Server solution, the use of an External Security Manager remains a fully supported option. In most cases, the decision to include such a component is based not only on the security of the immediate system, but on the value of deploying an “enterprise wide” calibre security solution. WebSphere Portal Server and the underlying WebSphere Application Server are secure in their own right and benefit less than one might expect from an External Security Manager. However, when many systems are consolidated within an organization, enterprise security adds significant value.

A maximum security policy would, however, dictate that additional security software adds to a solution by providing another layer that must be cracked; if not for anything else, then for fending off DoS (Denial of Service) attacks.

The implications of not architecting an External Security Manager, such as Tivoli Access Manager (TAM), should be fully understood. Most notably, without TAM, WebSphere Portal Server user accounts cannot be automatically locked after a certain number of invalid password attempts (three times) and concurrent logins using the same user account cannot be prevented. Unless, that is, a custom coding effort is undertaken to develop a Custom User Registry (CUR) for fulfilling such a purpose.

External Security Managers also address much larger problems, such as enterprise SSO (Single Sign-On), complex authentication, and centralized authorization.

2.6.3 Single Sign-On (SSO)

Single Sign-On (SSO) is the term used to describe a system or mechanism where users need to undergo a minimum number of explicit authentication steps in order to be given access to multiple systems or services. SSO enhances user convenience by automating access to all authorized servers and services through a single authentication process. This capability eliminates the need to remember multiple sign-on processes, user IDs, or passwords. Moreover, by this single action, user authentication errors are reduced.

The purpose of SSO is to:

- Provide a SSO capability for all Web-based applications. A user should only need to log in one time to one entity to obtain access to all authorized applications and content, which may reside on various servers.
- Provide a centralized point of authentication, generating a valid credential (ticket, cookie, and so on).
- Remove the need for application developers to specifically authenticate users within their application code. The intricacies of security can be abstracted from such applications.
- Provide a cross-platform security solution. Experience has shown that there is a need to maintain operating system independence for Web-based application security.
- Provide the ability to control access to Web applications and content, which may be hosted through multiple Web servers, at the URL level.
- Provide the ability to make fine-grained authorization decisions within applications. While this is not an immediate deployment requirement, the solution must allow for this capability to be added.
- Support browser based access to applications from both customers and employees. From their desks, internal users may access both internet-hosted applications and internal applications. At this time, there is no requirement for employees to have access to internal applications from the internet.

What SSO is not:

- An Identity Management Solution.
- A Federated Identity Management Solution.

Out-of-the-box SSO with WebSphere Portal Server

WebSphere Portal Server, or rather the underlying WebSphere Application Server instance, provides SSO functionality out-of-the-box. However, it is important to understand the capabilities and constraints associated with such a deployment. This statement is made in as much that the out-of-the-box SSO functionality may be insufficient for some enterprise-wide implementations, but also in the context that the adoption of an External Security Manager may simply be overkill.

Key points to note about the out-of-the-box SSO provided with WebSphere Portal Server are:

- SSO is based on the Lightweight Third-Party Authentication (LTPA) token, which is an IBM proprietary standard. It is suitable for achieving SSO between WebSphere and Domino based products only.

- SSO is a function of the underlying WebSphere Application Server instance. As such, there is no concept of a Reverse Authenticating Proxy Server, which could otherwise be place in a DMZ for added security.
- Pseudo-SSO is achieveable with the use of the Credential Vault. However, a user is required to manually enter his or her user ID and password prior to accessing the back-end system, as the user registries are typically not synchronized.
- SSO functionality does not extend to any fancy password expiry or user session handling. That is, concurrent logins using the same user account are not barred.

Enterprise SSO with an External Security Manager

The decision, therefore, to deploy an External Security Manager for a given implementation is usually based on a number of factors. However, one main requirement that often dictates the inclusion of such a product is the demand for an enterprise-wide SSO capability. As mentioned previously, Tivoli Access Manager is just one such product that represents the IBM strategic enterprise-wide security offering. TAM consists of two main components: the Policy Server and the WebSEAL Reverse Authenticating Proxy server. That is, when a user logs into a WebSphere Portal Server solution protected by TAM, it is actually the Tivoli WebSEAL server that performs the authentication task.

As such, the key points for deciding to deploy TAM above the out-of-the-box SSO provided by WebSphere Portal Server, are listed below:

- TAM provides enterprise-wide SSO capabilities.
- Basic Authentication SSO support.
- Forms-based SSO (FSSO) support.
- Lightweight Third-Party Authentication (LTPA) SSO support.
- HTTP Header based SSO support.
- Global SSO support.
- SPNEGO (Desktop SSO) support.

And in addition, the following aspects are provided:

- Centralized administration at an organizational level.
- Expired password handling.
- Password reset and password strength policy management.
- Delegated security administration for portal.
- Session duration or inactivity timeout.
- Account lockout (possibly for a specified period of time) after a specific number of successful authentication attempts.

Attention: It should be noted that the deployment of an External Security Manager, such as Tivoli Access Manager, does not necessarily address every aspect of SSO. For example, SSO is generally considered to be homogenous between all participants in a solution. Should the participants in a solution utilize different user repositories, there may well be the need to deploy an Identity Management Solution or a Federated Identity Management Solution.

Single sign-off

One often neglected aspect of SSO is the allied sign-off or sign out action associated with a user session. This is especially important because it is not uncommon for the back-end servers participating in the SSO realm to create and issue their own authentication cookies as part of the transparent SSO process. Unfortunately, if a user explicitly logs out of WebSEAL, but does not close or terminate the browser session, the back-end server cookies will remain active within the browser session. This, then creates a security threat by raising the risk that an unauthorized user may gain access to restrictive information, although it should be acknowledged that this kind of threat is only applicable to shared desktops and workstations, such as kiosks.

To overcome this threat, the recommendation is to embed JavaScript code capable of searching and destroying all applicable cookies in the page that WebSEAL redirects to after logging out a user.

2.6.4 Trust Association with WebSEAL

In this configuration, the underlying WebSphere Application Server instance of WebSphere Portal Server needs to be configured to explicitly "trust" the WebSEAL server so that if WebSEAL has already authenticated a user, WebSphere Application Server will not challenge the user to authenticate again. WebSphere Application Server provides a Trust Association Interceptor (TAI) framework for this purpose. Based on the established trust, WebSphere Application Server can map the delegated credential from WebSEAL to a valid WebSphere Application Server credential. The identification of the user must be passed to the TAI in a header called iv-user, which is inserted by WebSEAL into the HTTP Header of the request sent from WebSEAL to the WebSphere Application Server. Note that while WebSEAL can be configured to pass the user identity in other ways, the iv-user header is the only one supported by the TAI. Also note that the user password is not passed in the HTTP Header (for security reasons). After the TAI processing is successful, WebSphere Application Server creates a user authentication cookie called an LTPA token (we recommend using the LTPA token2). This is identical to the process that occurs when the out-of-the-box SSO with WebSphere Portal Server is enabled.

With the availability of the new TAI++, TAI now does not query the User Registry (LDAP) directly for Trust Association Interceptor processing. Instead, the new Interceptor class TAMTrustAssociationInterceptorplus contacts the Tivoli Access Manager (TAM) Authorization Server, which then proceeds to check the User Registry (LDAP). The benefit that this brings to a solution is that the TAM Authorization Server does not have to be the actual TAM Policy Server. As such, a local TAM Authorization Server replica can be installed, either alongside each WebSphere Portal Server node, or on an additional number of dedicated servers. As one might expect, this requires an additional amount of effort and planning, not to mention system resources. However, this is very beneficial when overall performance is a concern, as requests can be offloaded and cached at the TAM Authorization Server replicas, which may or may not be local to each WebSphere Portal Server node, greatly improving performance. After successful processing, TAI++ adds the PDPrincipal to the WebSphere subject or context, which can be retrieved by downstream applications. Furthermore, with the new TAI++, one can add custom attributes to the subject or context in the form of Java sets.

After the TAI++ has accepted the user identification and WebSphere Application Server has created the LTPA token, the WebSphere Member Manager component of WebSphere Portal Server performs additional queries against the User Registry (LDAP). In particular, WebSphere Member Manager does an LDAP search to get group and additional attribute information from the LDAP. WebSphere Portal Server also queries the resource mappings from the Portal database, before displaying the applicable Portal pages.

All communication should be over SSL; the link from WebSEAL to the Web server must use client certificate authentication, and the same must be true for the link from the Web server to the embedded Web Container of the underlying WebSphere Application Server instance of WebSphere Portal Server. Should either link be insecure, the TAI will continue to work, but the link will not be secure.

2.6.5 LTPA token generation with WebSEAL

With this option, one does not need to configure the Trust Association Interceptor (TAI) framework in WebSphere Application Server at all. Instead, one configures an LTPA junction in WebSEAL, and Tivoli Access Manager issues the LTPA token. The junction from WebSEAL to the WebSphere Application Server is configured to pass the iv-user and iv-groups information, and the LTPA token that is created by TAM. At the WebSphere Application Server, TAI is not enabled and the Application Server simply receives the LTPA token in the HTTP header request. The underlying WebSphere Application Server only creates the session cookie for the user and assumes that this user has already been authenticated.

It should be noted that just like the approach outlined in 2.6.4, “Trust Association with WebSEAL” on page 40, the WebSphere Member Manager component of WebSphere Portal Server must perform an addition LDAP query to retrieve a further number of user attributes and to determine the group membership for the concerned user.

2.6.6 Other Tivoli Access Manager considerations

The following recommendations are made with respect to integrating Tivoli Access Manager V6.0, both the Policy Server and WebSEAL components, with WebSphere Portal Server V6.0.1.

WebSphere Portal Server login with Tivoli WebSEAL

Most WebSphere Portal Server deployments include a number of anonymously accessible Portal pages. Advanced configurations may even see the default Portal Login page replaced with a Login Portlet. When WebSphere Portal Server is used in conjuction with Tivoli Access Manager, special consideration needs to be exercised to ensure that both products work in unison.

By default, the standard WebSEAL configuration is to intercept each new client request and to challenge a user to sign-in against a login form before accessing, through a proxy, the actual requested content. Unfortunately, what is not clearly documented in many publications is that this login form is actually the WebSEAL login page. Furthermore, unlike a WebSphere Portal Server page, the WebSEAL page is only capable of supporting static HTML content. Potentially, this page could be customized to mimic the same look and feel as the Portal. However, this may then lead to inconsistencies if WebSEAL is deployed as an enterprise wide security solution.

An alternative approach to the above would be to configure WebSEAL to allow unauthenticated requests to reach, through a proxy, the anonymous pages of WebSphere Portal Server. In this manner, it would be possible to use WebSphere Portal Server to display dynamic Portlet based anonymous content. A user could also interact with a Login Portlet when attempting to register and authenticate him or herself into the solution. In such a situation, the Login Portlet would simply post the supplied user ID/password to the WebSEAL pkmslogin Servlet. The post command may, or may not, include a string containing the URL of the page to redirect to after successful authentication.

WebSEAL high availability

The failure or outage, either scheduled or unscheduled, of a WebSEAL server will result in the need for a user to re-authenticate unless a suitable mechanism is configured to handle such conditions. Load balancer affinity normally ensures that subsequent requests from a client go to the same WebSEAL server for enhanced performance. However, when a WebSEAL server fails, the Load Balancer will redirect the user's request to the next available WebSEAL server. However, the WebSEAL-to-user session is maintained on an individual WebSEAL cache basis and when one such WebSEAL goes down, all other WebSEAL servers treat this as a new request and require re-authentication to be done by the user.

To overcome these issues, and to achieve seamless failover between multiple WebSEAL server, one of two mechanisms can be deployed:

- The Failover Cookie approach is a mechanism by which an additional cookie is created (and updated every time) and sent by WebSEAL during authentication to the user. When a request is failed over to another WebSEAL server, that server decrypts the cookie to understand that the user was a pre-authenticated against the initial WebSEAL server.
- The Session Management Server, newly introduced in TAM V6.0, is a useful alternative for maintaining the failover sessions of WebSEAL in a persistent manner. As such, the Session Management Server exposes a Web service to persist the user sessions to a database.

WebSEAL load balancing

WebSEAL includes the built-in capability for providing load balancing and failover when two or more back-end systems participate in the same junction definition. However, it is important to recognize that such a configuration does not extend to gracefully quiescing user requests, from one or more back-end systems, when those systems need to be taken down for scheduled maintenance. This is in contrast to the feature rich functionality provided by most commercially available load balancers. The requirement to be able to gracefully quiesce users, from one or more back-end systems, is especially important when deploying a dual clustered WebSphere Portal Server architecture with Two Lines of Production.

Session Management Server

The Session Management Server, as discussed previously in “WebSEAL high availability” on page 42, is a newly introduced feature of Tivoli Access Manager V6. In addition to the ability to handle session failover between multiple WebSEAL servers, the Session Management Server can also be deployed to restrict the number of concurrent logins or sessions each user can have at one time. Such a requirement is often in demand when WebSphere Portal Server is deployed in the Financial Services sector.

However, using the Session Management Server requires additional resources, as the component runs as a WebSphere Application Server based application. Furthermore, at the current time of the writing this document, the Session Management Server does not support any database server other than DB2.

Common Auditing and Reporting Service (CARS)

Tivoli Access Manager Version 6 also includes the new IBM Common Auditing and Reporting Service (CARS) platform, which provides a consistent way to collect audit events and report on the collected data. However, like the newly included Session Management Server, the CARS event server also demands consideration during the early stages of a project, as the component also runs as a WebSphere Application Server based application. Likewise, at the current time of the writing this document, the component only supports DB2 for data storage.

One may wish to consider CARS as an alternative to exploiting the generic UNIX syslogd for centrally collecting audit events in a distributed environment, as the standard syslogd does not provide encryption or any guarantee of delivery by being based on UDP.

2.6.7 LDAP Directory Servers

There are several aspects to LDAP Directory Server design that make the topic a non-trivial issue. Two of the most important aspects are described below.

LDAP directory structure

There are potentially a number of issues and considerations concerning the structure of the Directory Information Tree (DIT) when using WebSphere Portal Server, particularly when an existing populated LDAP directory is required to be used or when a new structure is to be defined from scratch.

DIT example

The suffix of an LDAP directory server is usually defined as part of the installation and configuration process. In the example illustrated in Figure 2-5 on page 44, the suffix has been fixed as dc=uk, dc=acme, dc=com, which adheres to the domain name syntax-based convention. Potentially, this could be revised to just dc=acme, dc=com or even dc=acme, dc=co, dc=uk.

It is anticipated that a number of organizational units (OU) would be needed at the topmost level to provide a degree of granular isolation between subordinate categories. As such, ou=people and ou=groups are normally created. It is intended that ou=people will contain all user entries and that the ou=groups will contain all the subordinate sub-groups that relate to the various functional departments of an organization.

The ou=people organizational unit directly contains the many user identities for the Portal solution. The hierarchy is totally flat with no boundaries between the users. The distinction is not made as to which department or Line of Business (LOB) a users belong under ou=people. Instead, the ou=groups organizational unit contains further sub-organizational units representing the different departments of an organization, such as ou=GroupA or ou=GroupB. This approach allows for greater flexibility when a user is assigned to work in a new department and so on.

Users are required to be associated with a group depending on their Portal "Role". Membership of a specific group therefore maps to a specific Portal "Role" and determines what access the user will be privileged to experience.

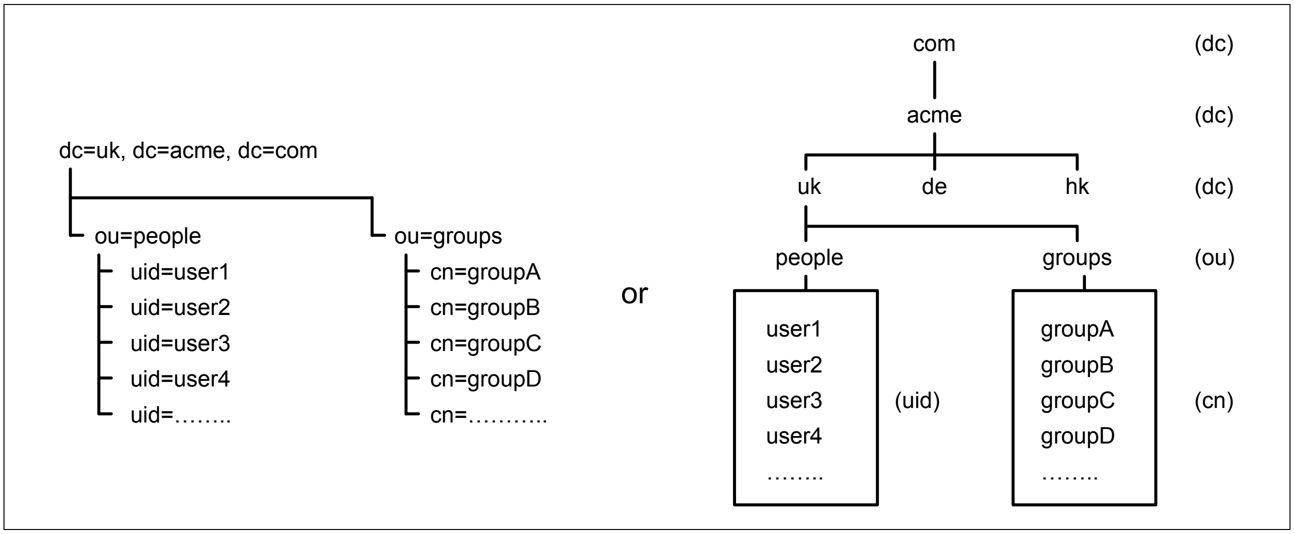


Figure 2-5 LDAP Basic DIT Design

LDAP schema design

By default, the WebSphere Portal Server configuration assumes that the underlying LDAP directory schema uses the object class applicable to the selected LDAP directory version, for example, InetOrgPerson when using IBM Tivoli Directory Server (TDS) V6.0. This is sufficient for most organizations, as it was defined to meet the requirements found in today's internet and intranet directory service deployments. However, in some cases it may not be sufficient enough. For example, it may be necessary to add the information of an employee's Account Number, Insurance Number, and Employment Band. These attributes do not exist in the standard InetOrgPerson object class.

Modifying the default object class, in an attempt to add or change an attribute, is not recommended. If the definition of one of the default attributes, for example, givenName, needs to be changed, then we recommend that a new attribute be created. However, such an attribute should only ever be created in a new custom object class. Objects can be derived from other objects. This is known as sub classing. An object class of AbcPerson could be defined as a subclass of the inetOrgPerson object class. The AbcPerson object class would have the same attributes as the inetOrgPerson object class and could add other attributes such as Account Number, Insurance Number, and Employment Band. This prevents potential conflicts when a new version of the directory is installed and the default schema is refreshed. One special object class, called top, has no superiors. The top object class includes the mandatory object Class attribute. Therefore, the attributes in top object class appear in all directory entries.

LDAP directory server selection

Make no mistake, all LDAP directory servers are not created equal. Tivoli Directory Server (TDS) was designed as standards-compliant enterprise directory server from inception. One of the main strengths that TDS has over other directories is that data is retained in an underlying DB2 database. Here, the DB2 database engine provides scalability to tens of millions of entries, as well as groups of hundreds of thousands of members. When this alone is compared to directories that store data as metadata on a file system, there is a distinct performance and integrity advantage.

The Lotus Domino LDAP implementation only supports the indirect method to locate the group memberships for a user. As such, it is not possible to determine the group membership of a given user by querying the user object directly. Instead, group membership is achieved by iteratively searching through the member list of all groups. A second limitation of the Lotus Domino LDAP implementation is that the number of members in a group is limited by the size of the field. To work around this issue, nested groups can be implemented, whereby members are divided across two or more groups and then each of these groups are added as members to the original group. Unfortunately, both these limitations impact the amount of time it takes to perform the Portal login step. For situations where large LDAP deployments have been configured within excess of 900 groups and 80,000 users, it is commonly acknowledged that the Portal login action will take a longer than usual time.

LDAP directory server high availability

WebSphere Portal Server V6.0.x introduced support for multiple LDAP directory servers with respect to new multi-realm capabilities. Not surprisingly, this has lead to some confusion when deploying multiple LDAP directory servers in response to the requirements of high-availability. As such, when multiple LDAP directory servers are deployed in support of a multi-realm deployment, often used in conjuction with Virtual Portals, these LDAP directory servers need to be highly available in their own right.

For Tivoli Directory Server based implementations, high availability is achieveable through the deployment of two directory servers that operate in a master peer-to-peer topology. However, in a slight deviation from the standard peer-to-peer practice, which works on a concept that there are multiple master peers in an environment each being capable of processing read and write requests, the recommend solution is to utilize a load balancer to preference one master peer as the active member for all read and write requests. The reason for this decision is to eliminate any potential conflicts that would otherwise result from two-way replication.

As such, the load balancer should be configured to always route read and write requests to the nominated master peer during normal operation. However, should the load balancer detect a failure of the master peer, the load balancer will re-route all requests to the alternate master peer. During write requests, there will be replication from 'node 1' to 'node 2', not the other way round, as there should not be any write requests being distributed across both LDAP servers or peers. It follows that read only requests can be evenly distributed to both peer LDAP servers. This can be achieved by configuring a second load balancer cluster group, with a different virtual host name to make a distinction from the first load balancer cluster group and virtual host name.

Note: It is not implied that by deploying a load balancer as a mechanism for handling LDAP directory server failover that it is possible to distinguish between the actual read and write requests of a particular application. For example, this technique does not imply that it is possible to determine which requests originating from WebSphere Portal Server are of a read nature and which are of a write nature, on a per request basis.

For those software products that include built-in LDAP redundancy, such as the Authorization Server and WebSEAL components of Tivoli Access Manager, there is no requirement for a dedicated load balancer. Moreover, the inclusion of a load balancer could impact the ability of the built-in fail-over mechanism to work effective.

It is not uncommon for the same load balancer, as mentioned above, to also serve a critical part in the overall solution architecture. That is, the load balancer or, more accurately, the back-end load balancer, is responsible for load balancing the many requests that originate from the WebSphere Portal Server cluster members to the various back-end servers. For example, it should be apparent that user requests do not bypass Portal Server to directly access the various back-end servers. Rather, it is the actual Portlet applications deployed within WebSphere Portal Server that invoke the services provided by the back-end servers. A Portal page, as such, may aggregate the response from several back-end servers. In such circumstances, it is important to ensure that the load balancer itself does not become a bottleneck, as this will have the potential to impact the overall performance of WebSphere Portal Server.

When using LDAP over SSL (LDAPS), care should be taken when utilizing a load balancer as described above. LDAPS not only establishes a JNDI context against the target server, but also implements SSL handshaking between the client and target server (including key negotiation). Whether the load balancer simply just redirects the SSL connection to the target directory server or whether the SSL connection is terminated at the load balancer, with the load balancer re-negotiating a secondary SSL connection to the target directory server, needs to be decided.

LDAP directory servers and firewalls

Problems can arise if a firewall is placed between WebSphere Portal Server and the chosen LDAP directory server. Under such circumstances, authentication can appear to stall after a long period of inactivity. This typically manifests itself in the morning after a night of inactivity, whereupon users may wait up to 30 minutes before authenticating into the Portal solution (unless the Portal is restarted or the LDAP Reuse Connection parameter is disabled from the WebSphere administrative console and WMM connection pooling mechanism is disabled). After this initial period, subsequent users are authenticated in the normal fashion.

The origin of this problem is not with WebSphere Portal Server or the underlying WebSphere Application Server instance, but with the firewall idle timeout. System Administrators should ensure that the tcp_keepidle system setting on each of the servers is smaller than the firewall idle timeout. Failing this, when a client is left to idle for longer than the firewall idle timeout, a communications error will be encountered. Usually, a keepAlive packet is sent according to the tcp setting of tcp_keepidle.

2.7 Database considerations

The deployment of a suitable database is probably one of the single most critical factors in a WebSphere Portal Server implementation. As several choices and combinations are available, selecting the most optimal architecture involves careful consideration of many factors.

2.7.1 WebSphere Portal Server database disclaimer

Details of the actual underlying data storage layouts are abstracted and hidden from the WebSphere Portal Server administrator. The WebSphere Portal Server schema is not published and IBM reserves the right to make modifications to the schema in future Portal Fix Packs. Any manual manipulation of the underlying data store is strongly discouraged, to the point that it will not be supported by IBM.

2.7.2 Database domains

With the introduction of database domains in WebSphere Portal Server V6.0.x, greater flexibility was made possible in terms of the permissible operational architecture. A full description of each of the database domains can be found in the WebSphere Portal Server Version 6.0 Information Center. In addition, further information including worked examples can be found in the *WebSphere Portal Version 6 Enterprise Scale Deployment Best Practices*, SG24-7387, found at:

http://www.redbooks.ibm.com/abstracts/sg247387.html

In this section, we provide a high-level overview of the two of the most common deployment options.

The dual cluster with Two Lines of Production architecture

Figure 2-6 depicts a dual clustered WebSphere Portal Server V6.0.x architecture supporting “Two Lines of Production”. Each “Line of Production” consists of multiple WebSphere Portal Server cluster members and accesses that are effectively the same community, customization. and wmm database domains. Such domains are said to be “shared” and are responsible for ensuring a consistent user experience. The sharing of the database domains ensures that data is automatically available to both “Lines of Production”. The one exception to this is the release database domain associated with each “Line of Production”, as each release domain maintains unique data specific to that “Line of Production” or cluster.

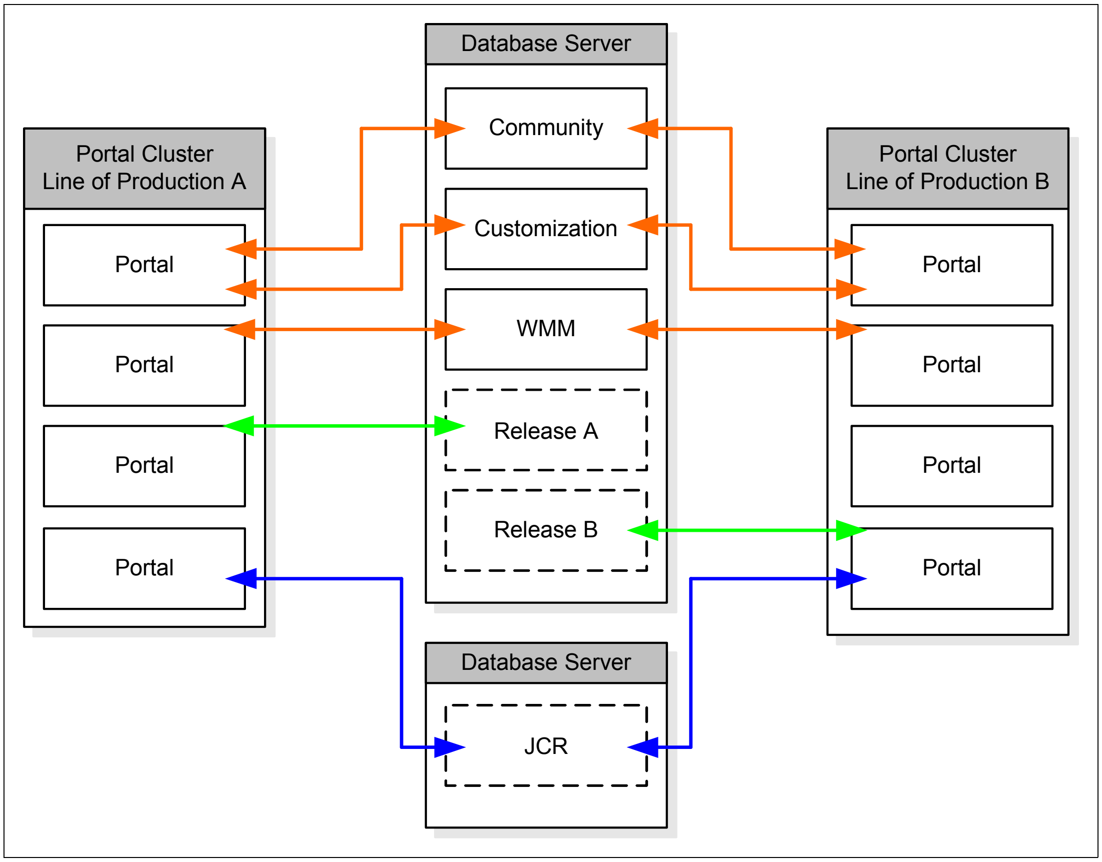


Figure 2-6 Database domains in a dual clustered WebSphere Portal Server V6.0.x architecture

Any user customization made against one cluster member, regardless of the “Line of Production” or cluster, by a user, is now available to the same user, as and when that user accesses any of the other cluster members participating in the same or different “Line of Production”. It should be acknowledged, however, that under normal conditions session affinity is maintained against the same cluster member for the duration of a user's session. The only exception to this is when a cluster member becomes unavailable, either through a deliberate or an unscheduled outage.

Figure 2-6 also shows a separate database server containing a shared JCR database. Although not mandatory, it should be acknowledged that a JCR database will have very different growth rates and performance characteristics from the main Portal domain databases. Architecting a separate database, as such, possibly in a different database instance or on a different physical database server, is one common option.

The geographically deployed architecture

In a geographically deployed WebSphere Portal Server V6.0.x architecture, as shown in Figure 2-7, each geography maintains its own set of databases. Each database would be highly available in its own right. However, the subtle difference between this implementation and that shown in Figure 2-6 on page 47 is that the shared database domains are replicated across a Wide Area Network (WAN) using such techniques as queue replication or 2-way SQL replication.

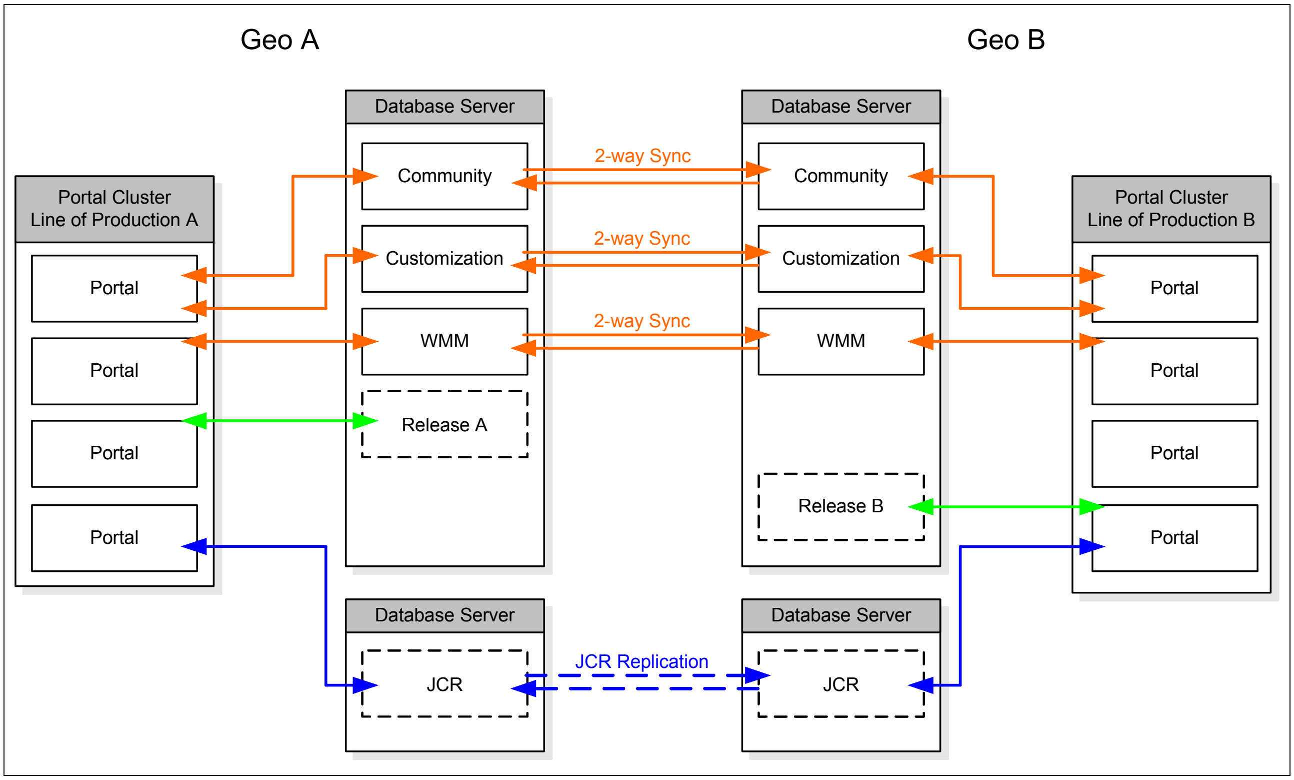


Figure 2-7 Database domains in a geographically deployed WebSphere Portal Server architecture

> **Important:** The implementation of a multi-clustered WebSphere Portal Server V6.0.x architecture that sees the deployment of individual clusters in each geography to support a truly "global deployment" mandates the use of the same LDAP directory server in all geographies. The LDAP directory server may, however, be replicated for redundancy purposes. This requirement is necessary to maintain uniqueness between Portal users.

2.7.3 Distinct databases or distinct schemas

The WebSphere Portal Server architecture allows each of the required database domains to exist in the same database instance. However, for availability and performance reasons, it is strongly recommended that due diligence is performed. If, for example, one database domain has different access characteristics and growth rates, differentiating between distinct databases would allow any DBA to specifically tune and size that database accordingly.

A DB2 instance is a logical database server environment. DB2 databases are created within DB2 instances on the database server. The creation of multiple instances on the same physical server provides a unique database server environment for each environment or sub-system. For example, the primary WebSphere Portal Server instance and the

WebSphere Portal Server instance associated with a stand-alone WCM deployment can be managed on the same machine in isolation.

> **Tip:** For those organizations using Oracle as their preferred database, the DB2 terminology described in the WebSphere Portal Server Version 6.0 Information Center can lead to confusion. In Oracle terms, a WebSphere Portal Server database domain should be considered an Oracle schema. As such, multiple schemas can exist within the same Oracle database. However, for the reasons outlined previously, it may on occasion prove beneficial to architect a separate Oracle database for a particular schema, for example, when considering the JCR Repository requirements of WCM and PDM.

2.7.4 Database high availability

To safeguard against catastrophic failure of the proposed WebSphere Portal Server solution, it is essential that the database tier is highly available. When using DB2, two principle methods exist for achieving high availability.

High Availability Cluster Multiprocessing

High Availability Cluster Multiprocessing (HACMP™), can be used to implement hardware clustering. That is, HACMP can automatically switch over from a failing server to another server, thus minimizing unscheduled down time. As such, the HACMP software detects that there is a problem with the initially active node and initiates the following actions on the standby node:

- Take over the applicable IP addresses.
- Take over the shared disks.
- Start the necessary application processes.

This is commonly known as a cold-standby configuration; only one node is actively running workload at a time. Furthermore, it is important to recognize that HACMP does not offer data redundancy.

Appendix B, “7x24 Maintenance” in the *HACMP for AIX 5L V5.2 Administration and Troubleshooting Guide*, SC23-4862 has the most current and comprehensive information about maintaining a cluster in a 24X7 environment. It can be found along with the other documentation for HACMP at the following Web site:

http://www.ibm.com/servers/eserver/pseries/library/hacmp_docs.html

HADR

DB2 High Availability Disaster Recovery (HADR) provides a new alternative for delivering a high availability solution by replicating data from a source database, called the Primary, to a target database, called the Standby. HADR provides protection for both partial and complete site failures. Combined with the new Automatic Client Reroute (ACR) capability, HADR provides transparency to the application regardless of the failure type, from hardware, network, or software issues to disaster scenarios like fire. HADR provides multiple levels of protection allowing flexibility in the environment. Additionally, DB2 provides an easy to use wizard that allows the entire configuration to be set up in a matter of minutes.

HADR functionality is available as part of the DB2 UDB Enterprise Server Edition at no extra charge. Users of DB2 UDB Express and DB2 UDB Workgroup Server Editions can add HADR function to their servers by purchasing the DB2 UDB High Availability Disaster Recovery Option.

Figure 2-8 gives an overview of the DB2 HADR.

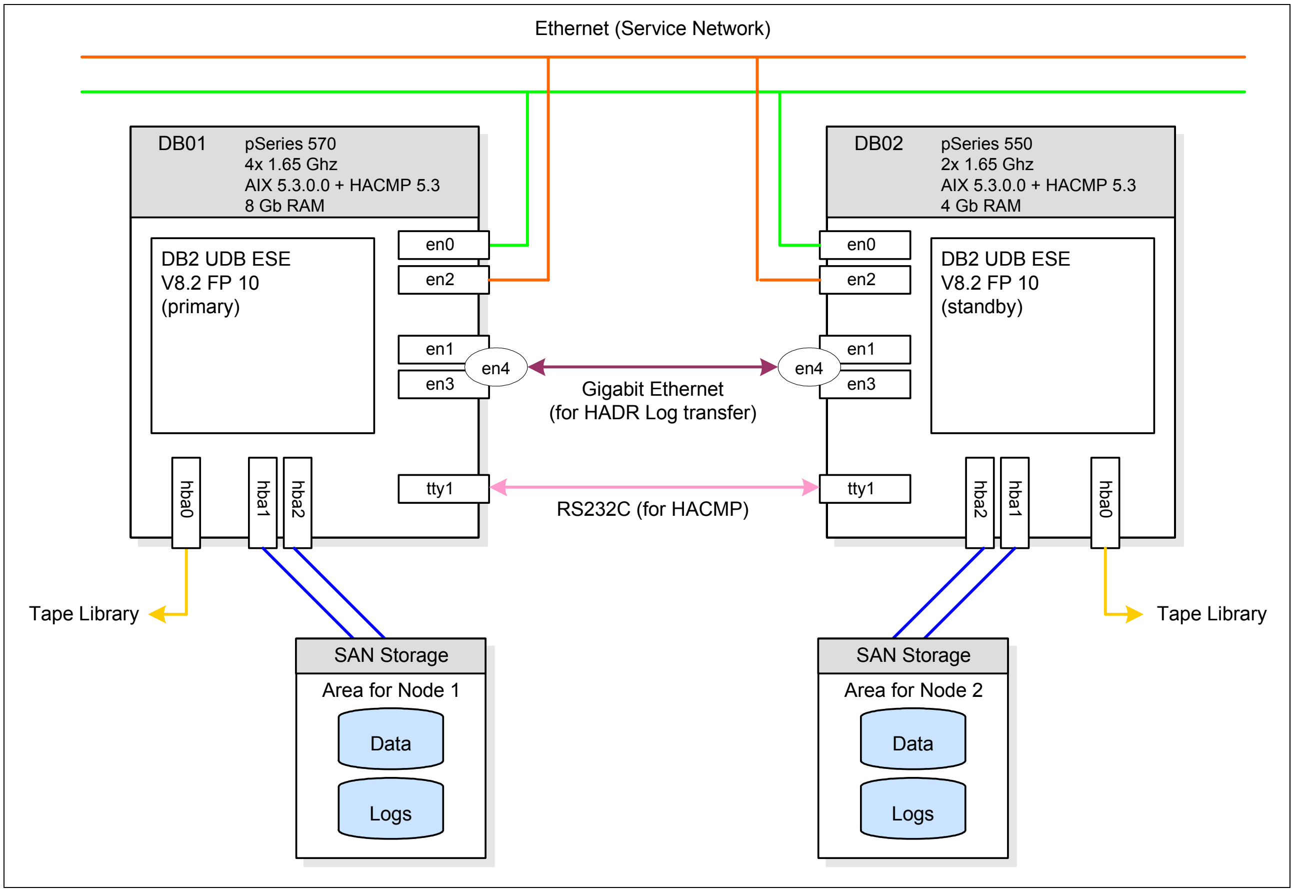


Figure 2-8 DB2 HADR

Without HADR, the length of time it takes to cut over from a database failure is unpredictable. It can take several minutes or hours before the failure is solved and the database is available. HADR enables failover and fallback between the two systems. The Standby database can take over as the Primary database with full DB2 functionality. After the failed old Primary is repaired, it can rejoin the HADR pair as a Standby database if the two copies of the database can be made consistent. After the original Primary database is reintegrated into the HADR pair as the Standby database, a failback operation can be performed so that the original Primary database is once again the Primary database. HADR requires the same hardware, OS, and DB2 software on the two systems (except for some minor differences).

A high availability mechanism is still a requirement, as DB2 HADR does not have a fault tolerant detection feature. In addition, if the Log transfer network is down, HADR takeover cannot be done. So this network is very important. In this configuration, a dedicated Gigabit Ethernet segment is used in conjunction with Network Interface Backup (NIB) for redundancy.

Note that an outage at the Log transfer network would cause the Primary to drop out of communication with the Standby (and if in Peer state, would cause the Primary to drop out of Peer state and run independent of the Standby). Once the network is repaired, the Primary and Standby would be able to eventually come back into Peer. Thus, a network outage at the Log transfer network would not result in a failure to process transactions, as seen by clients.

An alternative to HACMP is Tivoli System Automation (TSA). TSA is now bundled with DB2 for AIX (as of DB2 ESE 8.2 FP 13) in the same manner, and same licensing terms, as Linux (TSA bundled with DB2 ESE 8.2 on Linux). All HACMP or TSA has to do is detect a node failure and issue the TAKEOVER HADR command. There is no requirement to configure it to do any disk takeover, IP address takeover, or anything else, so the configuration is straightforward. When it detects that the Primary has failed, HACMP or TSA will run the TAKEOVER HADR ON DATABASE prod BY FORCE command, which will cause the Standby to become the Primary. Client requests are automatically redirected to the new Primary server using the Automatic Client Reroute (ACR) capability in the DB2 Client.

The real difference between instance failover and HADR failover is the time taken to be back up and running after a failure. With HADR, this can be under 30 seconds, but with HACMP instance failover, this is typically around one to two minutes.

Data redundancy is an additional benefit of deploying HADR when compared to HACMP instance failover alone. If the primary storage fails in a HADR configuration, failover can occur to the redundant storage. In the case of just HACMP, loss of the shared storage means catastrophic failure, with the only course of action being the restoration from a previous backup. Any incremental data will be lost.

2.8 Portal planning recommendations

As acknowledged at the beginning of this chapter, WebSphere Portal Server architectures come in many shapes and forms. A common requirement of any implementation, therefore, is the amount of attention and detail given to adequately planning such a project. Indeed, in order to minimize implementation risk, good planning is essential; failing to plan is planning to fail.

2.8.1 Recommendations for a successful implementation

We strongly recommend that a WebSphere Portal Server based implemention is treated as a complex infrastructure project from the outset. For anything other than an out-of-the-box implementation, which only encompasses laying down the core WebSphere Portal Server runtime, the complexity and length of time a project will need will grow significantly as more products or integration points are introduced. For example, an architecture incorporating an External Security Manager, such as Tivoli Access Manager, will demand extra resources with the appropriate skill set and additional time to both plan and implement.

In large WebSphere Portal Server solution projects, as with any other large scale based projects, it is crucial to have proper project management and governance mechanisms in place. Beside the large amount of work that is caused by the various workstreams, the demands in managing such undertakings are increased dramatically.

Assemble a multidisciplinary team

It takes a multidisciplinary team to successfully deploy a large scale WebSphere Portal Server implementation. As such, the two most important leaders on a delivery project are the Solution Architect and Project Manager. It should further be recognized that while the Solution Architect remains ultimately responsible for the overall solution design, it is possible that there are other architects under his or her command. For example, it is not uncommon that there is an architect assigned to each of the following categories: application, enterprise, integration, information, infrastructure, and operations.

Adoption of a methodology, pattern, or reference architecture

Although not mandated by any means, the adoption of a methodology, pattern, or reference architecture is strongly recommended when setting out on a WebSphere Portal Server project. For a complete listing of available patterns, consult the IBM Patterns for e-business Web site at:

http://www.ibm.com/developerworks/patterns

Adopt the Portal Build & Validate Methodology

In establishing a Portal Build & Validate Methodology, we acknowledge that there are key milestones associated with any Portal deployment. Adopting such a methodology thus reduces the likelihood that an incorrectly installed component will go undetected, until such a time that a significant Portal failure results. Our experience tells that among the most common causes of Portal deployment failures is the adoption of a big-bang approach. By contrast, the Portal Build & Validate Methodology breaks down the Portal deployment into discrete steps, each requiring validation. Failure to do so often results in ripping apart the solution, in an attempt to eliminate the various components of the architecture until the culprit is found.

Distinguish between COTS packages and proprietary code

With the availability of Commercially-Off-The-Shelf (COTS) packages, such as WebSphere Portal Server and Portlet applications that deliver specific functionality, the duration of a Portal implementation has been greatly reduced. However, it is important to recognize when custom development is needed, as in our experience Project Managers have not always been able to distinguish between COTS and custom developed applications.

Address performance as early as possible

Performance should be addressed as early on in a project as possible and then as an ongoing concern. All too often performance is disregarded until the performance tuning phase of a project, resulting in a critical situation. Consider performance testing those back-end systems prior to starting WebSphere Portal Server performance testing, as it is acknowledged that WebSphere Portal Server can never improve on the performance of any back-end system. Due to the iterative nature of performance tuning, no less than one month should be set aside for this important phase of any project.

> **Important:** Performance testing requires dedicated hardware and software. The expectation that performance testing can be performed using employee mobile computers or desktops is a serious misjudgment.

Include provisions for when things go wrong

On a regular basis, project teams neglect to make any provision for when things go wrong. A well thought out project plan includes such a provision. After all, it is far better to complete a deployment before the target date than to keep shifting the go-live target date. Plan on increasing any time set aside for this important facet of any project, when the level of complexity and the number of integrated systems increases.

Adopt a proper build mechanism

During the course of an implementation, there will be many versions of the components developed and deployed. As such, versioning is required when code and artifacts are promoted between the various environments of an implementation. For those deployments implementing a dual clustered architecture, with “Two Lines of Production”, it is especially important to have a proper build and deployment mechanism in place. This is to ensure that each line of production is identical, albeit when both lines of production are operational and not undergoing any incremental upgrade.

Deployment and cutover plan

Deployment can impose a great deal of change and stress for any organization. Therefore, ensuring a smooth deployment is a key factor in satisfying any stakeholder. A deployment and cutover plan, as such, should minimize the impact of the cutover with the stakeholder's staff, existing production systems and overall business routine. Creating a good deployment plan helps to identify most of the issues, vulnerabilities, and unforeseen glitches.

3

WebSphere Portal installation

This chapter contains information that will guide you through the installation of your WebSphere Portal Server. This chapter includes the following topics:

- Installation
- Database Transfer
- Enable Security
- Problem Determination

3.1 Installation

There is a great deal of information contained in this chapter so in an effort to prevent you from feeling overwhelmed, we recommend that you review the content that most relates to your environment.

3.1.1 How do I prepare my system for installation

This section highlights the minimum product levels that need to be installed before opening a problem report with the WebSphere Portal Support team. Because other products frequently ship fixes, updates, and new releases, testing every configuration is not possible. In general, you can install and run updates to supported products if those updates are forward compatible and are covered by the generic support statement found at:

http://publib.boulder.ibm.com/infocenter/wpdoc/v6r0/topic/com.ibm.wp.ent.doc/wpf/inst_req_supt.html

The list below identifies the supported products that work with the WebSphere Portal on the various operating systems. The supported product can run, but does not need to run, on the same machine or operating system where the WebSphere Portal runs. Check product-specific software requirements to determine whether the software runs native or connected to the WebSphere Portal.

Supported hardware and software

- WebSphere Portal V6.0 software requirements:

  http://publib.boulder.ibm.com/infocenter/wpdoc/v6r0/topic/com.ibm.wp.ent.doc/wpf/inst_req60.html

- WebSphere Portal V6.0.0.1 software requirements:

  http://publib.boulder.ibm.com/infocenter/wpdoc/v6r0/topic/com.ibm.wp.ent.doc/wpf/inst_req6001.html

- WebSphere Portal V6.0.1 software requirements:

  http://www-1.ibm.com/support/docview.wss?rs=688&uid=swg27009608

- WebSphere Portal V6.0.1.1 software requirements:

  http://www-1.ibm.com/support/docview.wss?rs=688&context=SSHRKX&context=SSBRWT&context=SSQKRQ&context=SS3NNG&context=SSYJ99&context=SSRUWN&context=SS6JVW&q1=software+requirements&uid=swg27010005&loc=en_US&cs=utf-8&lang=en

Prepare the operating system

This section includes information for setting up your operating system for WebSphere Portal with Cloudscape. Other components might require additional steps; see the product documentation for the specific components you want to install for information.

- Preparing an AIX machine:

  http://publib.boulder.ibm.com/infocenter/wpdoc/v6r0/topic/com.ibm.wp.ent.doc/wpf/os_aix.html

- Preparing an HP-UX machine:

  http://publib.boulder.ibm.com/infocenter/wpdoc/v6r0/topic/com.ibm.wp.ent.doc/wpf/os_hpux.html

- Preparing a Linux machine:

  http://publib.boulder.ibm.com/infocenter/wpdoc/v6r0/topic/com.ibm.wp.ent.doc/wpf/os_linux.html

- Preparing a Solaris machine:

  http://publib.boulder.ibm.com/infocenter/wpdoc/v6r0/topic/com.ibm.wp.ent.doc/wpf/os_solaris.html

- Preparing a Windows® machine:

  http://publib.boulder.ibm.com/infocenter/wpdoc/v6r0/topic/com.ibm.wp.ent.doc/wpf/os_win.html

- Preparing a System i5™ machine:

  http://publib.boulder.ibm.com/infocenter/wpdoc/v6r0/topic/com.ibm.wp.ent.doc/wpf/is_os_iseries.html

3.1.2 What is about to happen

The installation program provides a layered approach to building your environment by initially allowing you to start with WebSphere Portal as the base and then incrementally add other components.

Methods to install/uninstall

The following three methods are available to install or uninstall WebSphere Portal:

- Graphical User Interface (Installation Wizard):

  http://publib.boulder.ibm.com/infocenter/wpdoc/v6r0/topic/com.ibm.wp.ent.doc/wpf/conf_gui.html

- Console Interface:

  http://publib.boulder.ibm.com/infocenter/wpdoc/v6r0/topic/com.ibm.wp.ent.doc/wpf/conf_console.html

- Response File:

  http://publib.boulder.ibm.com/infocenter/wpdoc/v6r0/topic/com.ibm.wp.ent.doc/wpf/conf_response.html

Installation types

This section describes the different installation options for WebSphere Portal from the Quick installation scenario that gets WebSphere Portal up and running quickly with a completely integrated IBM Cloudscape database to the advanced installation scenarios that address special situations that might arise in your environment.

Stand-alone server

The most basic installation scenario is to install WebSphere Portal as a stand-alone server along with the WebSphere Application Server. If you currently do not have an existing WebSphere Application Server installed on your system, then this is the scenario you want to pursue. The typical installation sequence that occurs behind the scenes, which is traced in the wpsinstall.log, is as follows:

- Validation
  - Detects any currently installed versions of WebSphere Application Server.
  - Checks for the space requirements for WebSphere Application Server, Process Server, and for IHS (Web server).

  - Validates the operating system.
- Installs WebSphere Application Server base.
- WebSphere Application Server base is upgraded to V6.0.2.9.
- WebSphere Application Server fixes installed:

  http://www-1.ibm.com/support/docview.wss?rs=688&context=SSHRKX&context=SSBRWT&context=SSQKRQ&context=SS3NNG&context=SSYJ99&context=SSRUWN&context=SS6JVW&q1=Required+interim+fixes+for+WebSphere+Application+Server&uid=swg21245283&loc=en_US&cs=utf-8&lang=en
- Profile creation.
- WebSphere Portal Installation.
- Enable WebSphere Application Server Security.
- Configure BPE.
- Stop and start the WebSphere Portal Server.

For an in-depth analysis of the wpsinstall.log and for troubleshooting of the installation program, refer to Chapter 8, “Problem Determination”, in the *WebSphere Portal Version 6 Enterprise Scale Deployment Best Practices*, SG24-7387:

http://www.redbooks.ibm.com/abstracts/sg247387.html

Custom

A more custom type of installation is to install a new version of WebSphere Portal Server on an existing instance of WebSphere Application Server.

Once you launch the install program, you will select the **Custom install** radio button.

The installer will detect if you have one or more existing WebSphere Application Server instances installed and display them by the location. You must select the instance that you want to use from the list.

> **Tip:** If the installation program does not detect a WebSphere Application Server instance, but you know that it is present on the machine, exit the install and pass the location using the command line:
>
> ```
> ./install.sh -W was.undetectedWas="/my/WAS/location/"
> ```

Ensure the **Install on a managed node** option is checked when you are installing to a node that is already under deployment manager control. Also, you should already have a profile created if pursuing this installation option; however, if you do not, then you will need to specify all the WebSphere Application Server information in order for the install program to create the profile at this time.

The key difference between the custom installation scenario and the typical install is seen during the validation phase when the currently installed WebSphere Application Server version is detected and a check is done to see if any WebSphere Application Server instance was installed with Portal, as shown in Figure 3-1 on page 61.

Example 3-1 Custom Install trace output

```
(Jul 30, 2007 5:32:44 PM), MultiPlatform.install, com.ibm.wps.install.WasSelectPanel, msg2,
Number of currently installed WAS:1
(Jul 30, 2007 5:32:44 PM), MultiPlatform.install, com.ibm.wps.install.WasSelectPanel, msg2,
installed WAS 0: {Location=C:\IBM\WebSphere\AppServer, Version=6.0.2.19.0}
(Jul 30, 2007 5:32:48 PM), MultiPlatform.install, com.ibm.wps.install.WasSelectPanel, msg1, WAS
validation result for C:/IBM/WebSphere/AppServer: true
(Jul 30, 2007 5:32:48 PM), MultiPlatform.install,
com.ibm.wps.install.SetUserInputPanelPropertyAction, msg2, Attempting to set user input panel
bean 'was' property 'location' to 'C:\IBM\WebSphere\AppServer'
(Jul 30, 2007 5:32:48 PM), MultiPlatform.install,
com.ibm.wps.install.SetUserInputPanelPropertyAction, msg2, Setting bean property successful
(Jul 30, 2007 5:32:48 PM), MultiPlatform.install, com.ibm.wps.install.DetectWpsAction, msg2, WAS
Location: C:\IBM\WebSphere\AppServer
(Jul 30, 2007 5:32:48 PM), MultiPlatform.install, com.ibm.wps.install.DetectWpsAction, msg2,
Number of currently installed WPS:0
(Jul 30, 2007 5:32:48 PM), MultiPlatform.install, com.ibm.wps.install.DetectWpsAction, msg2, No WAS with
WPS detected.
```

After the system completes validation, the installer proceeds with the WebSphere Application Server profile creation, the WebSphere Portal Installation, and the enable security configuration task.

For more information regarding the step by step procedure for the custom installation scenario, refer to the WebSphere Portal Information Center topic “Installing with an existing instance of WebSphere Application Server”, found at:

http://publib.boulder.ibm.com/infocenter/wpdoc/v6r0/topic/com.ibm.wp.ent.doc/wpf/inst_wp_exwas.html

Co-existence installation

The co-existence installation scenario allows you to install more than one copy of WebSphere Portal on the same machine, where each server operates independently of the others. Each copy of WebSphere Portal is installed on a separate WebSphere Application Server profile and since all copies share the same system resources, such as processor capacity and memory, the multiple copies will impact performance.

When installing co-existing WebSphere Portal servers, you have the option to install each copy of WebSphere Portal with a new copy of WebSphere Application Server, as outlined in “Stand-alone server” on page 57, or to install each copy of WebSphere Portal on an existing version of WebSphere Portal, as in “Custom” on page 58.

Note: In order to avoid port conflicts with this installation scenario, you need to review the configuration methods outlined in the Information Center:

http://publib.boulder.ibm.com/infocenter/wpdoc/v6r0/topic/com.ibm.wp.ent.doc/wpf/inst_ports.html

Empty install

The empty portal installation scenario installs WebSphere Portal without the installation and deployment of default portlets and without the pages that are normally created with the typical and custom installation scenarios.

The empty portal installation is most often used when a transfer of the entire configuration, for example, from test environment to production environment is required.The XML configuration interface will allow you to export content from a test environment, for example, and import the content into a production environment.

The key point to remember in the empty portal installation is that the install program actually does removal of all WebSphere Portal resources, as noted in Example 3-2.

Example 3-2 Empty Portal Install trace output

```
(Jul 31, 2007 3:39:45 PM), MultiPlatform.install, com.ibm.wps.install.ExternalCommandAction,
msg2, Beginning install step: Removing all Portal resources
(Jul 31, 2007 3:39:45 PM), MultiPlatform.install, com.ibm.wps.install.ExternalCommandAction,
msg2, Executing command: cmd /c ""C:\IBM\WebSphere\PortalServer\config\WPSconfig.bat"
action-empty-portal -DPortalAdminPwd=PASSWORD_REMOVED -DWasPassword=PASSWORD_REMOVED
-DLTPAPassword=PASSWORD_REMOVED -DskipWTP=true"
(Jul 31, 2007 3:39:45 PM), MultiPlatform.install, com.ibm.wps.install.ExternalCommandAction,
msg2, Working directory:
```

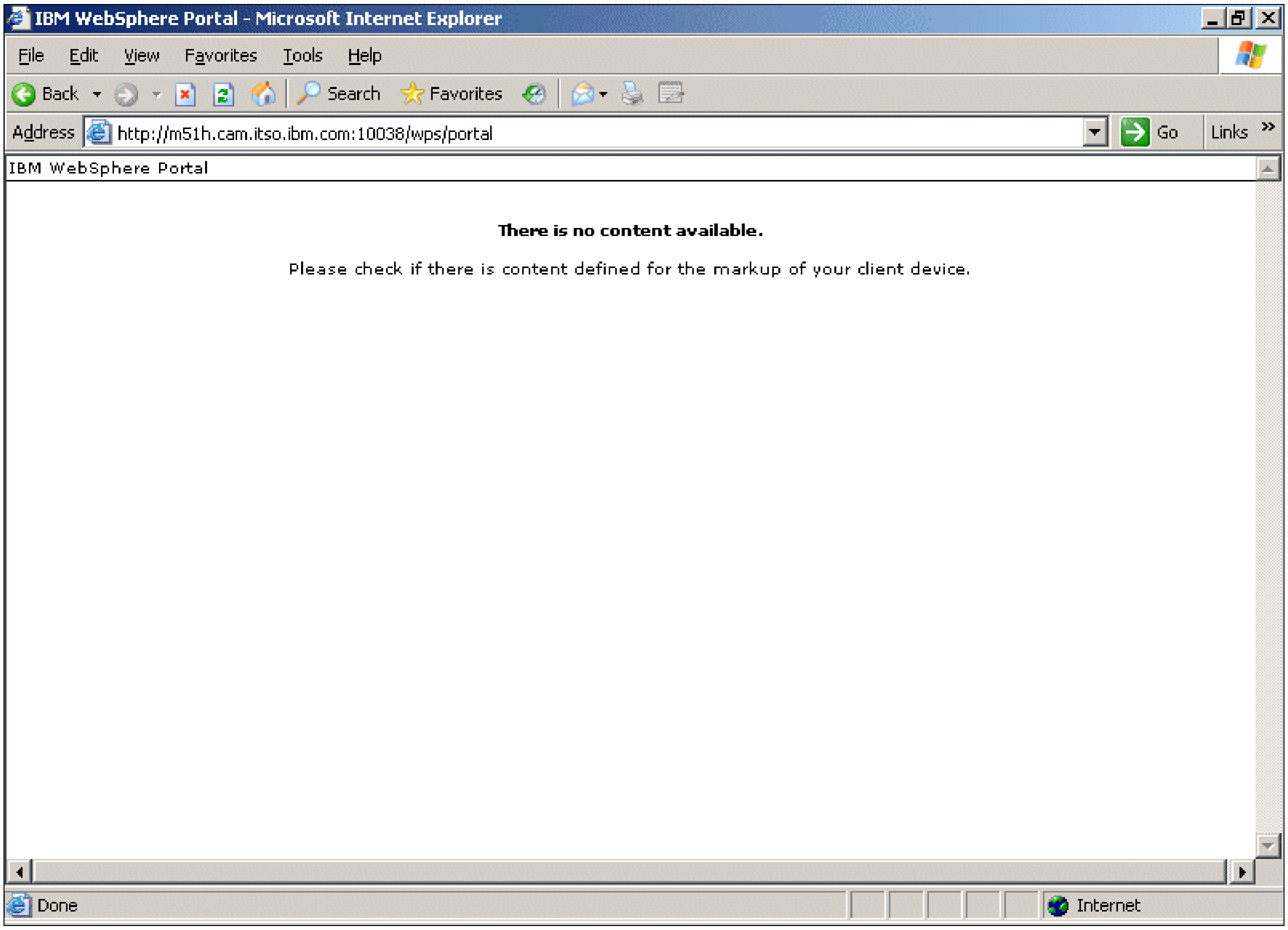
Upon completion of this task, you need to confirm the task completes with a BUILD SUCESSFUL message, as noted in Example 3-3.

Example 3-3 Empty Portal Install trace output

```
BUILD SUCCESSFUL
(Jul 31, 2007 3:47:07 PM), MultiPlatform.install,
com.ibm.wps.install.ExternalCommandAction$OutputWatcher, msg2, StdOut: Total time: 7 minutes 12
seconds
(Jul 31, 2007 3:47:08 PM), MultiPlatform.install, com.ibm.wps.install.ExternalCommandAction,
msg2, Return code = 0
(Jul 31, 2007 3:47:08 PM), MultiPlatform.install, com.ibm.wps.install.ExternalCommandAction,
msg2, Executing command: completed
(Jul 31, 2007 3:47:08 PM), MultiPlatform.install, com.ibm.wps.install.ExternalCommandAction,
msg2, Completed install step: Removing all Portal resources
```

Attention: After installing an empty portal, expect messages indicating that no content can be displayed or that some components are not yet configured. Adding content into the portal will prevent these messages from rendering.

Figure 3-1 on page 61 shows what the user should see when accessing the Portal Server directly after completing an empty portal installation scenario.


IBM WebSphere Portal - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
Address http://m51h.cam.itso.ibm.com:10038/wps/portal
Go Links
IBM WebSphere Portal
Done
Internet

There is no content available.

Please check if there is content defined for the markup of your client device.

Figure 3-1 Empty Portal default page

> **Attention:** IBM is no longer supporting the "action-empty-portal" to clean out the Portal system and build it up from there since this has resulted in too many unresolveable issues. This procedure goes outside of what "action-empty-portal" was intended for, so the "action-empty-portal" task is only to be used when preparing to import another Portal configuration. It is not to be used to clean out a Portal server for performance reasons or anything of that nature.

If you are attempting to improve Portal startup performance, you may try to stop applications that are not needed. Another area where startup performance (and memory usage) could be saved is to disable applications that may not be needed in your environment, such as Composite Applications and Templates (also known as "CAI/TAI") and Workplace Web Content Management™ (WCM). Some of the possible applications to stop are:

- ServletInvoker.war
- pickerPortlet.war
- JspServer.war
- mylist.war
- QuickLinks.war
- newsgroup.war
- docviewer.war
- FileServer.war
- reminder.war
- worldclock.war

- Bookmarks.war
- xslt.war
- sql.war
- CPPMail.war
- bannerad.war
- csv.war
- domdoc.war
- Exchange3.war
- MarketWatch.war
- WelcomePortlet.war
- Blurb.war
- SpellCheckerService.war
- LotusDocViewer.war
- Exchange2003.war
- QuickplaceInline.war
- LWP_CAI
- LWP_TAI
- content_j2ee
- dmdesktop
- icmjcrear
- ilwwcm_wcmsearchseed
- wcm

3.1.3 Where do I begin

Before you can begin the installation, you must choose the installation source that best fits your environment.

Installation source

The installation media for WebSphere Portal is distributed in two ways:

1. Electronically in the form of download images
2. Physically in the form of media CDs

Using discs to perform the installation is recommended if you have the CDs and plan on performing a limited number of installations, as little setup is required.

As an alternative to using discs, you can download WebSphere Portal software images onto a workstation or networked drive and then use those images to install the software. The primary delivery mechanism for retrieving the files necessary to install WebSphere Portal and its supporting software are the electronic Service Delivery (eSD) sites. These sites include Passport Advantage® and Partner World, which are linked directly to the IBM Customer Entitlement systems so visitors to the site will only see what they have purchased and are entitled to see. The files on the eSD site are arranged together in *e-Assemblies* reflecting the solution purchased by customers by platform. The term e-Assembly or e-Assy refers to a compilation of software images. Use one of these phrases to locate all the images that are packaged with the WebSphere Portal offering that you want to download. You can also search the software site for the part number of the product you wish to install rather then the e-Assembly. The part number can also be found in the TechNote mentioned in this section, for example, WebSphere Portal V6.0 - WebSphere Portal Enable Linux on x86, V6.0 eAssembly (CR45KML).

This is one of the choices that downloads the WebSphere Portal Enable V6.0 image for the Linux OS. If you are a customer using Passport Advantage to download images from the Web, or if you have access to the appropriate IBM internal software through Business Partner download sites, then refer to the WebSphere Portal V6.0 components outlined in this document:

http://www-1.ibm.com/support/docview.wss?rs=688&uid=swg24012969

Now that you have the appropriate e-Assembly or e-Assy image name, perform the following steps to locate images for download:

- Access the search mechanism on the Web site that you use to download software images.
  - IBM Passport Advantage:

    http://www.ibm.com/software/howtobuy/passportadvantage/pao_customers.htm
  - IBM Passport Advantage - Direct Link To Customer Login:

    https://www.ibm.com/software/howtobuy/passportadvantage/paocustomer
- Type the e-Assembly or e-Assy to locate the offering.
- View and accept the license terms.
- Select the correct images from the list of downloadable images that are displayed.

For steps on how to properly extract the CD images, refer to the "Choosing an installation source" topic in the WebSphere Portal Information Center, found at:

http://publib.boulder.ibm.com/infocenter/wpdoc/v6r0/topic/com.ibm.wp.ent.doc/wpf/inst_source.html

3.1.4 Is it working

In order to ensure a successful installation of WebSphere Portal, we recommend that the following steps are verified.

Accessing the WebSphere Portal URL

First, you will need to open a browser and access the Portal Server with the following URL:

```
http://example.com:port_number/wps/portal
```

where example.com is the fully qualified host name of the machine that is running WebSphere Portal and port_number is the port number that is displayed on the confirmation panel. For example:

```
http://www.ibm.com:10038/wps/portal
```

Logging into WebSphere Portal

Perform the following instructions for logging in to your portal:

- Click the **Log In** button in the banner (upper right hand corner).
- Type the administrator user ID and password in the appropriate fields since this is your first time logging into the WebSphere Portal Server after installation and additional users have not been created.

Note: Users must sign up to receive a user ID and password to log in to the portal.

- Click **Log in** to continue, or click **Cancel** to return to the default portal page.

For more information about Self Registration of users, refer to the WebSphere Portal Information Center section titled “Signing up to the Portal” and “Adding new Users” located at:

http://publib.boulder.ibm.com/infocenter/wpdoc/v6r0/topic/com.ibm.wp.ent.doc/wps/sign_up.html

wpsinstall.log

You will need to open the <wp_root>/log/wpsinstall.log and check for the following trace output, which confirms that the install and starting of WebSphere Portal has completed with no outstanding errors, as shown in Example 3-4

Example 3-4 wpsinstall.log trace output

```
(Jul 30, 2007 6:11:03 PM), MultiPlatform.install, com.ibm.wps.install.ExternalCommandAction$OutputWatcher,
msg2, StdOut: BUILD SUCCESSFUL
(Jul 30, 2007 6:11:03 PM), MultiPlatform.install,
com.ibm.wps.install.ExternalCommandAction$OutputWatcher, msg2, StdOut: Total time: 2 minutes 55
seconds
(Jul 30, 2007 6:11:04 PM), MultiPlatform.install, com.ibm.wps.install.ExternalCommandAction,
msg2, Return code = 0
(Jul 30, 2007 6:11:04 PM), MultiPlatform.install, com.ibm.wps.install.ExternalCommandAction,
msg2, Executing command: completed
(Jul 30, 2007 6:11:04 PM), MultiPlatform.install, com.ibm.wps.install.ExternalCommandAction,
msg2, Completed install step: Starting WebSphere Portal
```

ConfigTrace.log

Most commonly, the installation failures result from the configuration tasks that are executed during installation. The <wp_root>/log/ConfigTrace.log contains the generated trace output for each configuration task that executed during installation, so you will need to open this file and check for the output, as shown in Example 3-5.

Example 3-5 ConfigTrace.log trace output

```
start-portal-server:
   [logmsg] 2007.07.30 18:08:14.609 start-portal-server
   [logmsg]   EJPCA3163I: Starting Server "WebSphere_Portal"

     [echo] 'WebSphere_Portal' seems to be stopped.
     [echo] Starting 'WebSphere_Portal'
     [exec] ADMU0116I: Tool information is being logged in file
     [exec]
C:\ibm\WebSphere\profiles\wp_profile\logs\WebSphere_Portal\startServer.log
     [exec] ADMU0128I: Starting tool with the wp_profile profile
     [exec] ADMU3100I: Reading configuration for server: WebSphere_Portal
     [exec] ADMU3200I: Server launched. Waiting for initialization status.
     [exec] ADMU3000I: Server WebSphere_Portal open for e-business; process id is 2228
Target finished: start-portal-server
Mon Jul 30 18:11:03 EDT 2007
Target started: action-post-config
action-post-config:
   [delete] Deleting: C:\IBM\WEBSPH~1\PORTAL~1\config\work\was\wp_portal.properties
   [delete] Deleting: C:\IBM\WEBSPH~1\PORTAL~1\config\wpconfig_ascii.properties
Target finished: action-post-config---- Begin dump of properties ----
* listing of all properties are seen in this section *
---- End dump of properties ----
BUILD SUCCESSFUL
```

SystemOut.log

The loading of WebSphere Portal begins with the trace output, as shown in Example 3-6.

Example 3-6 SystemOut.log trace output

```
[7/30/07 18:09:03:781 EDT] 00000016 WebGroup      A   SRVE0169I: Loading Web Module: WebSphere Portal
Server.
[7/30/07 18:09:04:219 EDT] 00000016 WebApp        A   SRVE0180I: [WebSphere Portal Server]
[/wps] [Servlet.LOG]: ServiceManager: Loading from
file:/C:/IBM/WebSphere/PortalServer/shared/app/config/services.properties
[7/30/07 18:09:04:266 EDT] 00000016 LogManagerDef I com.ibm.wps.logging.LogManagerDefaultImpl
init
   --------------------------------------------------------------------------------
   IBM WebSphere Portal 6.0

   Licensed Materials - Property of IBM
   5724-E76 and 5724-E77
   (C) Copyright IBM Corp. 2001, 2006 - All Rights Reserved.
   US Government Users Restricted Rights - Use, duplication or disclosure restricted by GSA ADP
Schedule Contract with IBM Corp.
   --------------------------------------------------------------------------------
   Build Level:
   wp600_244 (2006-07-18 17:02)
   --------------------------------------------------------------------------------
which follows with the confirmtaion that WebSphere Portal has been initialized :
[7/30/07 18:09:33:578 EDT] 00000016 ServletWrappe A   SRVE0242I: [wps] [/wps] [portal]:
Initialization successful.
```

The remaining lines of output will indicate the loading and initialization of all the WebSphere Portal applications. You will need to verify that the applications are loaded and initialized with no errors. Directly after this information, you should see the “Server WebSphere_Portal open for e-business”, which confirms your WebSphere Portal Server is now up and running.

3.2 Database transfer

By default, WebSphere Portal Server automatically installs and stores its predefined data in the IBM Cloudscape Database, as shown in Figure 3-2. While the IBM Cloudscape Database may be the suitable choice in small scale deployments, organizations looking to leverage the enterprise-wide capacity attributes of a database management system should continue with the following sections.

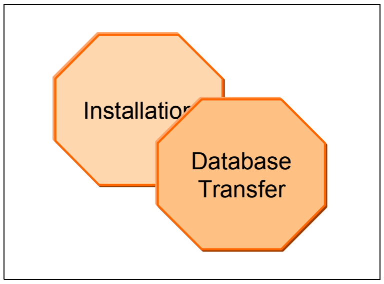


Figure 3-2 Database transfer

3.2.1 Planning and considerations

WebSphere Portal V6 provides new options to address scalability and redundancy in your enterprise deployments. If you choose to transfer to an external database, we recommend that you do so before you add a large amount of content and preferably during the installation and configuration process. Figure 3-3 shows some points to consider before you transfer you data to a database of enterprise scale.

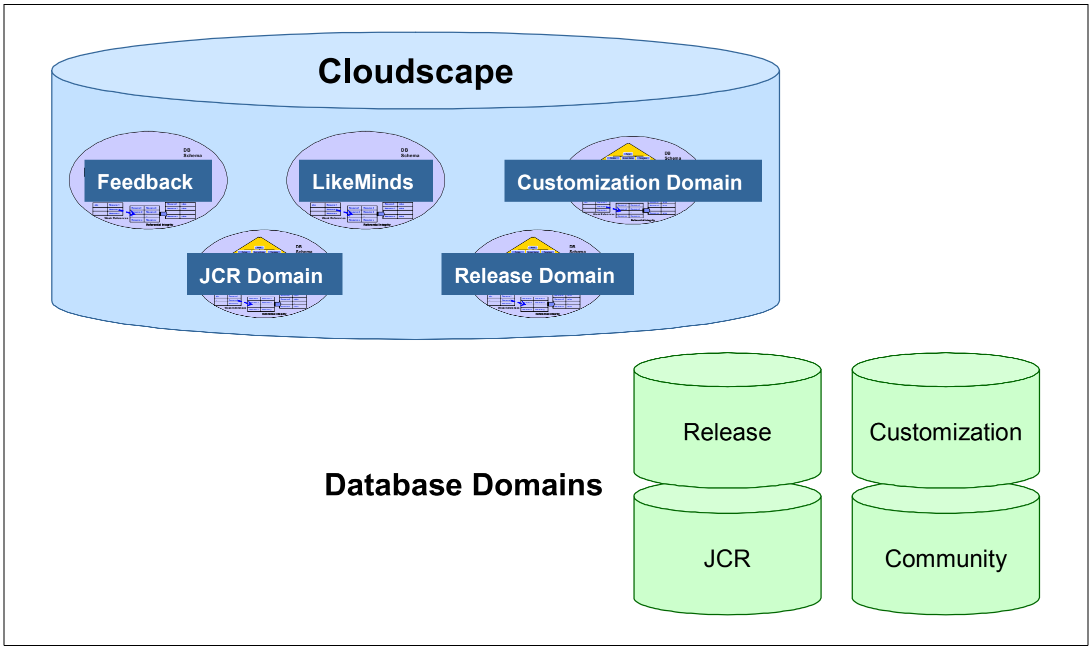


Figure 3-3 Transferring to an external database

Audit of your database infrastructure

The database is typically the heart of any Web-based application. For the success of your deployment, it is critical that the hardware and software that will be used to house the portal database(s) be not only highly optimized but resilient. Before you begin the database transfer process, review the prerequisites discussed below.

System requirements

It is important to conduct a preliminary review of the hardware and software of your system in both the new and existing database infrastructures to ensure they meet the supported levels for WebSphere Portal Server. The InfoCenter is routinely updated with specific versions and recommended compatible levels of configuration. If your are considering an upgrade to your database implementation, we advise you to refer to 3.1.1, “How do I prepare my system for installation” on page 56 *before* attempting an upgrade of your environment.

Performance and availability

WebSphere Portal Server provides you with the option of installing the database server on the same server that the WebSphere Portal Server will be housed; however, if performance is of utmost importance for your portal application(s), we recommend that you provide a separate physical database server for your RDBMS. Performance tools should be utilized continuously to monitor the state of the database server(s) and the databases themselves, with mechanisms instituted to tune the entities as needed.

For additional information about planning for your database infrastructure, refer to 2.7, “Database considerations” on page 46.

Database domains

With WebSphere Portal Server V6, the content repository has been separated into database domains. The separation of domains increases the flexibility for organizations by permitting:

- Single instances of WebSphere Portal Server to share portal data without clustering.
- Sharing of portal data among portal clusters allowing for multiple lines of production, allowing organizations to comply with high availability requirements.
- Partition of portal data not just among database management systems, but database software types (that is, DB2 and Oracle)

.

> **Note:** Release, LikeMinds, and Feedback Data cannot be shared between databases. The Sync and Designer Databases are supposed to stay on Cloudscape and therefore are not part of the database transfer process.

While you have the liberty in WebSphere Portal Server V5.1 to create all portal data under one database for performance, availability, and scalability purposes, we recommend that you create separate physical databases for each of your database domains.

Authorization

WebSphere Portal Server does not require an ID with plenipotentiary authority on the server(s) that will house your external portal database(s). For the purposes of system to system communication, we recommend that you create a user on your database servers and assign to the group those user who have been granted DBADM authority (or its equivalent) in your database management system.

3.2.2 How do I prepare for the database transfer

To prepare for the transfer of your database(s) from Cloudscape to an external database, you should execute the following steps:

1. If you have not done so already, the first thing you should do before attempting to transfer your portal data is to make a file system level backup of the portal server(s) you have installed up to this point.
2. After the file system backup has completed, make a secondary backup of your wpconfig.properties, wpconfig_dbdomain.properties, and wpconfig_dbtype.properties files located in your WP_root/config directory. These files will be modified for the database transfer process, so it is good to have a secondary backup readily available without having to have the files restored.

> **Note:** You should always create a backup of files at each step of the installation process and at any point in time when the files will need to be modified.

3. Whether your database management system is installed locally (on the same system as WebSphere Portal Server) or remote, the installation and configuration of your database server (s) should be complete and your database servers should be tuned for optimal performance, as noted in 3.2.1, “Planning and considerations” on page 66. The database admin ID that WebSphere Portal Server will use to access data should be created and assigned to the appropriate database administrative group.

4. If you are connecting to an external database remotely, create the database(s) you plan to utilize as instructed in the InfoCenter instructions. Users of DB2 have the convenience of having WebSphere Portal Server create the databases locally by running `./WPSconfig.sh/WPSconfig.bat create-local-database-db2` from the command line.
5. WebSphere Portal Server V6 allows for the use of the Type 4 JDBC™ driver for all supported databases, eliminating the need to install a local database client on the WebSphere Portal Server for those distributed environments. Before you begin the database transfer process, copy the required Type 4 jar file(s) for your database management system server over to your WebSphere Portal Server(s). Should you choose to have the local client installed for remote connection (required for supported platforms in which will use the Type 2 JDBC driver), you should install the database client on all WebSphere Portal Servers beforehand. Catalog the databases on the WebSphere Portal machine. Note that this is not required for those environments in which the database server will be installed locally. For more information about driver support, refer to "Supported hardware and software" on page 56.
6. For most platforms, you have the option of transferring the database manually using the command line, or transferring the database using the configuration wizard. Regardless of the process you choose, you will need to modify the wpconfig_dbdomain.properties, wpconfig_dbtype.properties, and wpconfig.properties with the values required in order to perform the database transfer, as both methods will pull the information from these files. Do not provide values for other parameters in the properties files other than those specified in the InfoCenter instructions.

Note: If you are planning to use the optional LookAside feature in your portal implementation, do not set this value during the database transfer process. This value is set as a part of enabling security.

7. Verify the database connections from WebSphere Portal Server to your database(s) by running the validate database connection configuration tasks for the individual domains you will be transferring to your external database. If you receive failures, do not continue with the additional steps until the tasks run successfully.
8. Check the InfoCenter to re-confirm that you have followed all instructions for your Database Management system, including any system requirements or other modifications necessary for your database management system. You can access the InfoCenter at:

   http://publib.boulder.ibm.com/infocenter/wpdoc/v6r0/index.jsp
9. Using support tools *before* you run installation and configuration tasks are another effective method of problem avoidance. By utilizing a tool such as IBM Support Assistant you can perform a search for "database transfer with WebSphere Portal Version 6" and review all TechNotes surrounding this topic. Refer to Appendix A, "Using IBM tools to find solutions and promote customer self-help" on page 169 for more information.
10. Stop WebSphere Portal Server if it is running. In a clustered environment, Deployment Manager and *all node agents should be running and synchronized*.

Table 3-1 on page 69 gives you a checklist of all the required items for the database transfers preparation.

Table 3-1 Database Transfer Preparation Checklist

Step	Task	Completed
1	Install, configure, and tune your database management system.	□
2	Assign an ID or privilege that will be used by WebSphere Portal Server(s) for system to system communications from the portal to the database.	□
3	Create the WebSphere Portal database(s) on the database server.	□
4	If you are using the Type 4 JDBC driver, copy the required files from your database server to your WebSphere Portal Server(s).	□
5	If you are using the Type 2 JDBC driver, install the database client on the WebSphere Portal Server(s).	□
6	Make a file system backup of WebSphere Portal Server.	□
7	Make a secondary backup of wpconfig_properties, wpconfig_dbdomain.properties, and wpconfig_dbtype.properties.	□
8	Enter database values in wpconfig.properties, wpconfig_dbdomain.properties, and wpconfig_dbtype.properties.	□
9	Validate the database configuration by running the Validate database connection tasks.	□
10	Check the InfoCenter to reconfirm that you have met all the prerequisites for hardware, software, and configuration for your database server and client.	□
11	Problem Avoidance: Use the IBM Support Assistant to perform a search on "database transfer and WebSphere Portal V6" to review all TechNotes and solutions associated with this task.	□
12	Stop all WebSphere Portal Server(s): In a clustered environment, verify the Deployment Manager is started and all node agents are running and synchronized.	□

3.2.3 What is about to happen

We recommend that you perform the database transfer before you use WebSphere Portal extensively if you choose to transfer data to another supported database, since large amounts of data in the databases can cause the database transfer to fail if your Java heap size is not large enough. Since data is added to the database as you use WebSphere Portal, you should perform the database transfer as soon as it is practical to do so to avoid problems due to the amount of data you are transferring.

You can transfer data from any supported database type to any other supported database type, which means the source database does not have to be the default database that is created during installation. In order to transfer data between supported databases, you must edit the wpconfig_sourceDb.properties file and update it with the source databases' information. This file is a copy of wpconfig_dbdomain.properties that is created automatically during installation.

Attention: Data can be transferred from a Cloudscape database, but cannot be transferred to a Cloudscape database, so if you are transferring from a database other than the default database, you will need to edit the wpconfig_sourceDb.properties file. This file is a copy of the wpconfig_dbdomain.properties file, so the default values will be for Cloudscape if it is not modified.

If you want to transfer your data to another supported database, you will need to follow the steps specific to the type of database you are using, for example, DB2, Oracle, or SQL Server™. By this point, you should have planned for the database you wish to use, installed the database for use with WebSphere Portal, and set up the database to work with WebSphere Portal that is, creating users and creating local or remote databases. Once these steps have been followed (per 3.2.2, “How do I prepare for the database transfer” on page 67), you should be ready to transfer the data.

You may choose to transfer data manually or by using the configuration wizard and you may transfer all the data at once or by individual domains. When transferring data manually, you want to first validate the configuration properties with the following task:

- For UNIX:

```
./WPSconfig.sh validate-database-connection-wps -Drelease.DbPassword=password
-Dcustomization.DbPassword=password -Dcommunity.DbPassword=password
-Djcr.DbPassword=password
./WPSconfig.sh validate-database-connection-jcr -Djcr.DbPassword=password
./WPSconfig.sh validate-database-connection-feedback
-Dfeedback.DbPassword=password
./WPSconfig.sh validate-database-connection-likeminds
-Dlikeminds.DbPassword=password
./WPSconfig.sh validate-database-connection-wmm -Dwmm.DbPassword=password
./WPSconfig.sh validate-database-driver
```

- Windows:

```
WPSconfig.bat validate-database-connection-wps -Drelease.DbPassword=password
-Dcustomization.DbPassword=password -Dcommunity.DbPassword=password
-Djcr.DbPassword=password
WPSconfig.bat validate-database-connection-jcr -Djcr.DbPassword=password
WPSconfig.bat validate-database-connection-feedback
-Dfeedback.DbPassword=password
WPSconfig.bat validate-database-connection-likeminds
-Dlikeminds.DbPassword=password
WPSconfig.bat validate-database-connection-wmm -Dwmm.DbPassword=password
WPSconfig.bat validate-database-driver
```

This task will begin by validating a connection with each domain by running a separate set of subtasks for each one. Once this task is complete, you will need to stop the WebSphere Portal server and server1 and, upon stopping the servers, you will be ready to run the database transfer command as follows:

- For UNIX:

```
./WPSconfig.sh database-transfer
-Drelease.DbPassword=password-Dcustomization.DbPassword=password
-Dcommunity.DbPassword=password -Djcr.DbPassword=password
-Dwmm.DbPassword=password -Dfeedback.DbPassword=password
-Dlikeminds.DbPassword=password
```

- For Windows:

```
WPSconfig.bat database-transfer -Drelease.DbPassword=password
-Dcustomization.DbPassword=password-Dcommunity.DbPassword=password
-Djcr.DbPassword=password -Dwmm.DbPassword=password
-Dfeedback.DbPassword=password -Dlikeminds.DbPassword=password
```

The task will prepare for the configuration by deleting the existing /work directory, creating it, and then copying the relevant files for each domain into the directory. The task will then attempt to stop the WebSphere Portal server and admin server in order to proceed with the copying of the database properties. The task then begins the transfer part by attempting to make a connection to each domain using the jdbc providers. Once the connections have been validated, the task will proceed with a series of SQL scripts that first do a drop of each table and then a create of each table. At this point, all the tables should be created and ready for the transfer of data, which is performed next. If you view the ConfigTrace.log, at this point you will see the `Transfer started` output followed by a series of `Transferring table` traces that indicate the transfer of data from the default source database tables to the new destination database tables. This process repeats a few more times for each domain.

After running this task, a message indicating success, BUILD SUCESSFUL, should result. You should check the log ConfigTrace.log file to verify that this task was successful. If the configuration fails, verify the values in the wpconfig.properties, wpconfig_dbdomain.properties, wpconfig_sourceDb.properties, and wpconfig_dbtype.properties files, since problems with the transfer result mostly from incorrect values in these property files. Once you have corrected the values, then you may repeat this step. If the problem you are facing is not related to incorrect values and you wish to troubleshoot the exceptions, then refer to 3.4, “Problem determination” on page 80 for additional guidance.

3.2.4 Is it working

In WebSphere Portal V5.1 or earlier, one of the ways to verify the database connectivity was to test the JDBC connections using the WebSphere Application Server console or through the WebSphere Deployment Manager in a clustered environment. Due to architectural changes in WebSphere Portal Server V6, you cannot test the data sources successfully through either console right after the database transfer process has completed. Attempting to test the connection will fail, but this is not an indication of a problem with the database transfer process itself or with your WebSphere Portal Server Installation.

Here are the steps to verify that the transfer of your portal data from Cloudscape to an external database is successful. For clustered environments, the verification steps should in it ally be performed on the primary node. If you encounter problems during any of the steps in the recommended validation process below, refer to 3.4, “Problem determination” on page 80.

1. After the database transfer completes, you should receive a BUILD SUCCESSFUL message indicating that the transfer process has now completed. Review the ConfigTrace.log located in WP_root/log for any errors.
2. Shut down your WebSphere Portal Server and back up the SystemErr.log and SystemOut.log files located in the WP_root/log directory. Once the logs have been backed up, delete the existing SystemErr.log and SystemOut.log files so that fresh log files are created when the server is started. For those in a clustered environment, verify that all your nodeagents are in sync through the Deployment Manager. Restart the WebSphere Portal Server (the primary node only for a clustered environment) and check the SystemErr.log and SystemOut.log files to verify you do not receive any errors upon startup.

3. Once WebSphere Portal Server is up and running and you have verified that there are no errors, open a Web browser and direct it to one of the following URL (depending on your deployment configuration):
   - Single server deployment: `http://`*`hostname.example.com`*`:10038/wps/portal`
   - Cluster deployment: `http://`*`primarynode.example.com`*`:9080/wps/portal`

After that your are able to authenticate successfully, click the different links in the portal to make sure that no errors are received both in the browser and in the SystermErr.log and SystemOut.log files.

> **Note:** Again, do not proceed any further if you have encountered errors during the database transfer process. If you are not able to determine the cause of the error, refer to 3.4, "Problem determination" on page 80 for tips on how to pinpoint the problem.

3.3 Enable security

Figure 3-4 shows the LDAP transfer's part in the installation process.

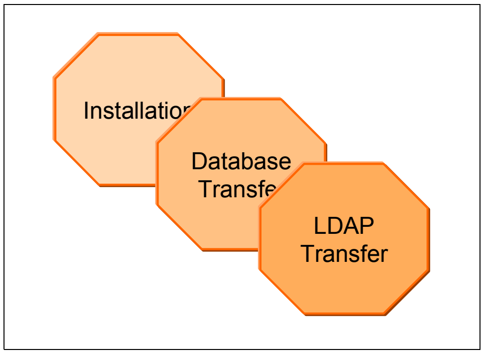


Figure 3-4 LDAP transfer

In this section, we will discuss the steps needed to connect your portal environment using an LDAP registry. The discussion focuses on enabling portal security with realms and without realms. The sections below provide a subset of points of consideration for your LDAP infrastructure, and restrict its focus to considerations before running the enable security targets. For a discussion on external authentication solutions, such as Tivoli Access Manager or Computer Associates eTrust Siteminder, as well as other topics surround LDAP planning, refer to 2.6.7, "LDAP Directory Servers" on page 43.

3.3.1 Planning and considerations

A number of organizations use LDAP to manage authentication and authorization for business applications. WebSphere Portal Server allows administrators to configure connections to an existing LDAP. The following sections highlight some of the important considerations that require review before connecting a portal to an LDAP user repository.

Audit of your LDAP infrastructure

The LDAP infrastructure should be reviewed and assessed before you connect WebSphere Portal Server to it. Like the database, the performance of the LDAP is vital to the usability of your portal and poor LDAP performance can render your portal inoperable. In this section, we discuss some inspection points that should be made.

System requirements

It is important to conduct a preliminary review of your system hardware and software in both new and existing LDAP infrastructures to ensure that they meet the supported levels for WebSphere Portal Server. The InfoCenter is routinely updated with specific versions and recommended compatible levels of configuration, If you are considering an upgrade to your LDAP implementation, we advise you to refer to 3.1.1, “How do I prepare my system for installation” on page 56 *before* attempting an upgrade of your environment.

Performance and availability

WebSphere Portal Server provides you with the option of installing the LDAP server on the same server that WebSphere Portal Server will be housed; however, if performance is of utmost importance for your portal application(s), we recommend that you provide a separate physical server for your LDAP.

- High Availability: Single LDAP servers provide a single point of failure and therefore are not a feasible option for deployment on an enterprise scale. For many environments, high availability is not a option or exception. The goal of high availability without performance impact are challenges organizations continue to face. High availability for the LDAP server is best achieved by having an LDAP proxy that will forward back-end requests. WebSphere Portal Server provides the option of configuring fail-over capability natively through the WebSphere Member Manager component. If you plan to configure WebSphere Portal Server for LDAP failover, you should enable security with realms and modify the wmm.xml as part of the post configuration steps in the InfoCenter. By default, the Reuse connection parameter should be enabled in the WebSphere Application Server console, or failover will not occur successfully should the primary server suffer an outage.
- LDAP Schema Design: While it is possible to set up WebSphere Portal Server with only one user and one group, this is not advisable. The LDAP Schema Design and Directory Information Tree (DIT) should ideally be thoughtfully planned and agreed to by all stake holders in your organization before you even attempt installation, and certainly before this phase in your deployment. Improper design of your LDAP Schema can affect the lookup performance in your LDAP, which will directly affect your portal implementation.
- Read-Only LDAP: LDAP uses existing users in your registry, meaning the users and groups will need to be created before they can access the portal. Authentication with read-only LDAP is performed using LDAP binding. Connection to a read-only LDAP WebSphere Portal Server requires an LDAP bind ID with the ability to read and search for the users in the subset of the DIT.
- LDAP that allow write permissions: Allows users to create and modify their personal attributes in a directory. When write access is allowed, WebSphere Portal Server users can use such features as Self Registration and self-care to register accounts for themselves. Write privileges to the LDAP requires an LDAP bind ID to be created with the ability to write and search for the users in the subset of the DIT.

> **Note:** In both instances, the LDAP Bind ID created for use with WebSphere Portal Server does not need to be the root ID for the directory server; in fact, it should not be.

LDAP Servers are oriented toward read-only operations and assume that information will be read from the LDAP server more than it is updated. Write operations will naturally be more expensive then read-only operations as a result and may require infrastructure changes to accommodate the cost. Review the documentation for your LDAP Server for discussion topics in this area.

- Filtering group information: The default filter information provided with your LDAP server is very generic in nature and geared toward searching and entire directory. Custom filters should be used to drill down to the subset of users in the LDAP tree to reduce the number of LDAP calls and improve overall performance of your portal.

LDAP security options

- Enabling a WebSphere Portal Server connection to an LDAP registry with realms

  Realms allow you to create group users from one or more LDAP Directory Information Trees and present them as a single entity to WebSphere Portal Server. Realms were introduced in WebSphere Portals Server Version 5.1, but support was limited to one registry. WebSphere Portal Sever V6 allows for the usage of multiple registries with realm enablement.

- Enabling WebSphere Portal Server connection to an LDAP Registry without realms

  When you enable security without realm support, only one user registry can be created. If your user information is contained in one LDAP, then you have the option of enabling security without realm support. For scalability and flexibility purposes, we recommend that you enable security with realm support.

> **Note:** At the time of the writing of this Redpaper, Web Content Management does not currently support WebSphere Portal Server environments with multiple realms. So you can either configure without realms or configure one realm in the WMM configuration files. Web Content Management is supported to use multiple registries, but they all need to be configured in the default realm. Planned support for multi-realms with WCM will be made available in a future release.

LookAside

LookAside is a repository that resides in the WebSphere Member Manager database. The purpose of LookAside is to provide the option to add additional attributes that do not correspond to a typical LDAP database. The LookAside option is available when configuring LDAP security with realms or without. Enabling LookAside can be done by setting the parameter `LookAside=true` in the wpconfig.properties file.

> **Note:** If you are planning to use Web Content Management, the LookAside database is required.

3.3.2 How do I prepare for WebSphere Portal Server LDAP security

The following presents the general steps you should take before you perform the enable security process.

1. LDAP installation, configuration and validation: The installation and configuration of your LDAP server should be completed by this phase. Performance tuning should be completed according to the recommendations in the LDAP server’s documentation and monitoring tools. A good way to test your LDAP configuration is to perform a search using the **`ldapsearch`** utility to confirm that your LDAP is operational.

   – Anonymous search:

   ```
   ldapsearch -s base -h ldaphostname “objectClass=*”
   ```

   – Using a Bind ID:

   ```
   ldapsearch -h ldaphostname -D “cn=wpsbind,o=co” -w “wpsbind” -s base
   “objectClass=*”
   ```

Note: Performing LDAP searches using an utility is one of the initial ways to troubleshoot directory problems. If you do not receive results and have confirmed that the problem is not user based (typos or extra spaces), it may indicate an underlying problem with the LDAP directory or network. Resolve these issues before proceeding with the enablement of directory security.

2. LDAP Design: While it is possible to set up WebSphere Portal Server with only one user and one group, this is not advisable. The LDAP Schema Design and Directory Information Tree (DIT) should ideally be thoughtfully planned and agreed to by all stake holders in your organization before you even attempt installation, and certainly before this phase in your deployment. Changing the LDAP Schema design during mid- or post-deployment could have unintended consequences.
3. LDAP requirements for WebSphere Portal Server: Before you can perform the connection to your user registry, the elements for user and group membership must be met. WebSphere Portal Server requires a minimum of one user and group to be created in LDAP before you can connect to it:
   - wpsadmin (Portal Administration User)
   - wpsadmins

   It is recommended that the wpsadmin user be a member of the wpsadmins group. If you are going to be using features such as Portal Document Manager (PDM) and WebSphere Content Management (WCM), we recommend creating the following groups ahead of time:
   - wasadmin (WebSphere Administration User; required if it will be different from the wpsadmin ID. The wasadmin user and wasadminGroup should be set ahead of time regardless of whether features such as PDM and WCM are utilized in your deployment.)
   - wpsContentAdminstrators
   - wpsDocReviewer
   - WcmAdminGroupId

Once all the users and groups are created, perform queries through the `ldapsearch` utility to validate the membership information used later to enable LDAP security.

4. Connectivity check (PING): From the server in which you will enable security, perform a ping test to verify the connection to your LDAP host(s). In addition to confirming that there is no packet loss, you should also verify that the round trip time is acceptable from destination to host based on your organization’s topology. Intermittent connectivity failures to your LDAP can cause not only your enablement of security task to fail, but can degrade the performance of WebSphere Portal Server. You should resolve all connectivity issues before attempting to run the enablement of security targets.
5. Connectivity check (TELNET): The next test that you should run from your WebSphere Portal Server(s) is to verify that you can telnet to the ports that are open and accessible from your LDAP Server(s). The default LDAP ports are 389 and 636. Port 636 is the default port used for Secure Socket Layer (SSL) connections. When performing the initial enablement of security for your user registry, the recommended sequence is to enable security connecting to the non-SSL port (389), then, after validating that your portal is able to connect to your LDAP successfully (link to the validation steps), follow the post configuration steps to connect your LDAP through SSL.

6. For most platforms, you have the option of enabling security manually using the command line, or transferring the database using the configuration wizard. Regardless of the process you choose, you will need to modify the wpconfig.properties and the helper file (optional) for your LDAP type. Take a secondary backup of these files before going through the enablement security process in order to avoid having to recover them from the file system backup.

Note: You should always create a backup of files at each step of the installation process and at any point in time where the files will need to be modified.

7. Stop WebSphere Portal Server if it is running. In a clustered environment, Deployment Manager and *all node agents should be running and synchronized*.
8. Check the InfoCenter to re-confirm that you have followed all instructions for your LDAP server, including any system requirements or other modifications necessary for your environment. You can access the InfoCenter at:

   `http://publib.boulder.ibm.com/infocenter/wpdoc/v6r0/index.jsp`
9. Using support tools *before* you run installation and configuration tasks are another effective method of problem avoidance. By utilizing a tool such as IBM Support Assistant, you can perform a search for “enable security with WebSphere Portal Version 6” and review all TechNotes surrounding this topic. Refer to Appendix A, “Using IBM tools to find solutions and promote customer self-help” on page 169 for more information.
10. Disable Security: Run the disable security task using the command line or the wizard. After the disable security task completes, you should receive a BUILD SUCCESSFUL message indicating that the transfer process has now completed. Review the ConfigTrace.log file located in WP_root/log for any errors. Start WebSphere Portal Server (if it is not already started) and verify that you can access the portal without credential validation.

Table 3-2 shows a checklist for the LDAP security preparation steps.

Table 3-2 LDAP security preparation checklist

Step	Tasks	Completed
1	Install, configure, and tune your LDAP Server(s).	□
2	Design your LDAP Schema and arrange the entries in the organization tree.	□
3	Perform a basic LDAP query using the **`ldapsearch`** utility to verify that your LDAP is functional.	□
4	Create wasadmin and wpsadmin (users) and wpsadmins (group) in your LDAP registry.	□
5	Perform a search for the wasadmin, wpsadmin, and the wpsadmins group by using the **`ldapsearch`** utility.	□
6	From your WebSphere Portal Server(s), perform an PING test to verify you connection to your LDAP host(s).	□
7	From your WebSphere Portal Server(s), perform a telnet test to port 389 (or 636 if 389 is closed) to confirm accessibility to the LDAP ports.	□
8	Make a file system backup of WebSphere Portal Server.	□
9	Make a backup of your portal data with the same time stamp as the WebSphere Portal Server(s).	□

Step	Tasks	Completed
10	Take a secondary backup of your wpconfig.properties file and the helper file (optional) for your LDAP Server.	□
11	Stop all WebSphere Portal Servers. In a clustered environment, verify that Deployment Manager is started and all node agents are running and synchronized.	□
12	Check the InfoCenter to reconfirm that you have met all prerequisites for hardware, software, and configuration for your LDAP, including the latest WebSphere Member Manager fixes.	□
13	Problem Avoidance: Use the IBM Support Assistant to perform a search for “enabling security and WebSphere Portal V6” to review all TechNotes and solutions associated with this task.	□
14	Proceed with the steps to disable security.	□

3.3.3 What is about to happen

After installation, WebSphere Portal Version 6 is installed with security enabled so the WebSphere Portal is functional right after installation and the configuration is suitable for a simple environment like unit tests or development. If you wish to configure a different type of security, other then the out-of-the-box configuration, then you have the option to configure WebSphere Portal to use a database registry, a custom registry, or an LDAP user registry to store user information and to authenticate users.

Although you can configure WebSphere Portal to use a database user registry to store user information and to authenticate users, this is not recommended for production scenarios due to performance constraints and the difficulty associated with managing a large number of users and groups. If still you wish to pursue this goal, you should refer to the InfoCenter, which discusses the issues to consider and the procedures to follow if you plan to use a database user registry as the WebSphere Application Server security type with WebSphere Portal. You can find the InfoCenter at:

http://publib.boulder.ibm.com/infocenter/wpdoc/v6r0/topic/com.ibm.wp.ent.doc/wpf/sec_was_en.html

The other option that provides considerable flexibility in adapting WebSphere Portal and security to various environments where some form of a user registry, other than LDAP or a Member Manager database, already exists in the operational environment is configuration of security with a custom user registry.

Attention: Implementing a custom user registry is a software development effort and will not be covered here.

At this point, you should be ready to configure security having WebSphere Portal installed and having completed the database transfer. In order to change the security configuration, you must first disable security and then re-enable it with the appropriate security configuration. Since installing an LDAP server is not part of the default WebSphere Portal installation, you must then install, set up, and configure the LDAP user registry. The InfoCenter outlines the procedures for each LDAP type, so reference the information to select the appropriate LDAP type to set up for your environment:

http://publib.boulder.ibm.com/infocenter/wpdoc/v6r0/topic/com.ibm.wp.ent.doc/wpf/ldap_u_reg.html

Once you have selected the type of LDAP for which you wish to configure security, proceed with the installation, the creation of required users and groups, the setup, disabling security, configuring, and verification of the LDAP.

> **Note:** Domino Directory is the LDAP user registry that will be discussed in this paper; you may reference the link above for information about the other types.

When configuring Domino Directory, you can choose to have one-realm support or multiple-realm support. Having realm support means one or more user registries (realms) can be created. Since a realm must be mapped to a Virtual Portal, having realm support will allow you to configure Virtual Portals in your environment as opposed to no realm support, which allows only one user registry (one realm) to be created. We recommend enabling security with realm support in the event that you wish to configure Virtual Portals in the environment at a later time.

The appropriate portal configuration tasks that are designed to configure portal security to work with an LDAP server can be run upon editing the entries in the file wpconfig.properties.

> **Note:** These instructions can apply to either a stand-alone server installation or a cluster environment. When performing these steps in a cluster, it is only necessary to perform them on the primary node.

For security purposes, we do not recommend that you store passwords in the wpconfig.properties file. Instead, you should edit the file with the passwords prior to running the configuration task and then remove the passwords from the file once the task has completed with success.

You also have the option to specify the password as a parameter when running the task through the command line, as shown in Example 3-7.

Example 3-7 Specifying the password as a parameter

```
WPSconfig.{sh|bat} task_name -Dpassword_property_key=password_value
```

Once you have located the wpconfig.properties file, you will need to create a backup before changing any values. Then using a text editor enter the values appropriate for your environment, such as the values in the following sections of the file:

- IBM WebSphere Application Server properties
- WebSphere Portal Security LTPA and SSO Configuration
- WebSphere Portal configuration properties
- LDAP Properties Configuration
- Advanced LDAP Configuration
- IBM Workplace Web Content Management Properties (If you wish to configure WCM in your WebSphere Portal environment)

You will also need to locate the wpconfig_dbdomain.properties file, which you will also need to create a backup of before changing any values. Then, using a text editor, enter the values appropriate for your environment, such as the values in the following sections of the file:

- Database properties

For detailed information about what needs to be added to the properties file, including examples, select **WebSphere Portal** → **Configuring** → **Configuring Security** → **LDAP**

user registry → Tivoli Directory Server/IBM SecureWay/Domino Directory/Active Directory/Novell eDirectory/Sun™ System Directory Server → Configuring (your specific LDAP user registry name here) → non-realm/realm support in the InfoCenter at:

http://publib.boulder.ibm.com/infocenter/wpdoc/v6r0/index.jsp

Once the files have been modified, you will need to save them and stop the WebSphere Portal server.

> **Note:** If this is a clustered environment, you will need to make sure the deployment manager and all the node agents are active.

To validate the configuration before enabling security, you can run the following tasks for:

- UNIX/Windows:

  ```
  ./WPSconfig.sh (bat) validate-wmmur-ldap -DWasPassword=password
  -DPortalAdminPwd=password -DLTPAPassword=password -DLDAPAdminPwd=password
  ```

- i5/OS®:

  ```
  WPSconfig.sh - profileName profile_root -DWasPassword=password
  -DPortalAdminPwd=password -DLTPAPassword=password -DLDAPAdminPwd=password
  ```

where *profile_root* is the name given to the WebSphere Application server profile in use.

> **Important:** Before enabling or disabling security in a clustered environment, using the appropriate tasks mentioned, you must stop all cluster members.

To configure WebSphere Portal security, you can run one of the following two tasks for UNIX/Windows:

- Realm Support:

  ```
  ./WPSconfig.sh(bat) enable-security-wmmur-ldap
  ```

- i5/OS:

  ```
  WPSconfig.sh -profileName profile_root -DPortalAdminPwd=password
  -DLTPAPassword=password -DLDAPAdminPwd=password
  ```

- Non-Realm Support:

  ```
  ./WPSconfig.sh(bat) enable-security-ldap
  ```

3.3.4 Is it working

Below are the steps listed to verify that portal security enablement using an LDAP user registry is successful. For clustered environments, the verification steps should in it ally be performed on the primary node. If you encounter problems during any of the steps in the recommended validation process below, refer to 3.4, “Problem determination” on page 80.

Verification

1. After the enabling security task completes, you should receive a BUILD SUCCESSFUL message indicating that the transfer process has now completed. Review the ConfigTrace.log file located in WP_root/log for any errors.

2. Shut down your WebSphere Portal Server and back up the SystemErr.log and SystemOut.log files located in the WP_root/log directory. Once the logs have been backed up, delete the existing SystemErr.log and SystemOut.log files so that fresh log files are created when the server is started. For those in a clustered environment, verify that all your nodeagents are in sync through the Deployment Manager. Restart the WebSphere Portal Server (primary node only for a clustered environment), and check the SystemErr.log and SystemOut.log files to verify that you do not receive any errors upon startup.
3. Once WebSphere Portal Server is up and running and you have verified that there are no errors, open a Web browser and direct it to one of the following URLs, depending on your deployment configuration:
   - Single server deployment: `http://`*`hostname.example.com`*`:10038/wps/portal`
   - Cluster deployment: `http://`*`primarynode.example.com`*`:9080/wps/portal`

After that your are able to authenticate successfully, click the different links in the portal to make sure that no errors are received both in the browser and in the SystermErr.log and SystemOut.log files.

If you configured your LDAP registry to allow users to create and modify attributes through features such as self-registration, create a user through self-registration and verify that the user appears in LDAP by using the **`ldapsearch`** utility.

3.4 Problem determination

Figure 3-5 shows problem determination's part in the installation process.

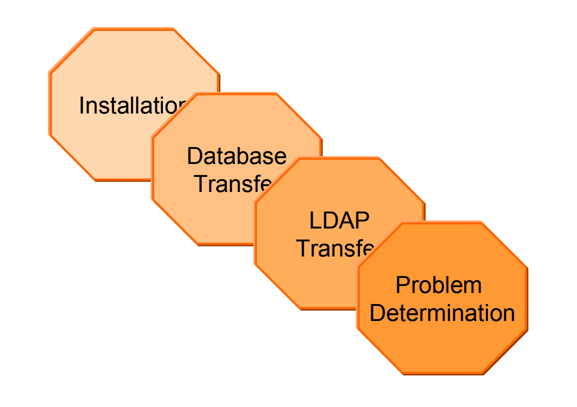


Figure 3-5 Problem determination

The following sections outline the recommended problem determination methods for solving issues related to WebSphere Portal installation as well as the database transfer and enabling security.

3.4.1 Installation problem determination

The most common installation failures occur during the configuration tasks that are run behind the scenes during the installation program. The initial sections of this chapter outline the contents of the wpsinstall.log for each installation scenario so you should be familiar with the events that take place during the installation of WebSphere Portal. This understanding should help you determine at what point the installation is failing to get a better idea of how to go about correcting the issue on your system. In addition to the wpsinstall.log, you will also need to review the logs that can be found in the system defined /temp directory or in the portal_root/log directory, which is created and the files copied into during the installation.

The recommended approach to troubleshooting an installation failure is start with the BUILD FAILED message output in the PortalServer/log/ConfigTrace.log file, which indicates a failure has occurred and analysis is needed. You will need to follow the general approach outlined in Chapter 8, “Troubleshooting and monitoring“, of *WebSphere Portal Version 6 Enterprise Scale Deployment Best Practices*, SG24-7387, found at:

http://www.redbooks.ibm.com/redbooks/SG247387/wwhelp/wwhimpl/js/html/wwhelp.htm

The general approach described in that IBM Redbooks publication can, if followed, help you determine the failures for many different error messages that can occur during the installation of WebSphere Portal.

3.4.2 Database transfer problem determination

This section shows some common problems with the database transfer process, and provides you with some ideas on how to solve them.

Incorrect Fix Pack levels

One of the most common causes of database transfer failures is not meeting the supported hardware or software requirements for your database management system. If you are installing a local client on the WebSphere Portal Server for remote connection to your databases, your client should match the same levels as your database server. If your server and clients are not at the required levels, refer to 3.1.1, “How do I prepare my system for installation” on page 56 and repeat the database transfer steps.

The fixes/Fix Pack issue is not isolated to the database servers. Not applying the required fixes/Fix Packs for your portal environment can cause errors during the database process and affect the operability of your portal environment.

Missing Jar file for Type 4 JDBC DRIVER

For database systems using the Type 4 JDBC Driver, you must copy the driver from the database server over to your WebSphere Portal Server. We recommend that if you have a local database client on your WebSphere Portal Server that you back up the jar file(s) created during the client install and replace them with the ones from the database server.

Incorrect entries in the wpconfig.properties files

This is perhaps the most common cause of database transfer errors. These types of errors are usually attributed to the following:

- Typos or extra spaces: Be certain to look over your properties files for misspellings and extra spaces. Ensure that the values entered are the same case throughout. Running the **`validate database connection targets`** before you conduct the database transfer may help you find some of the errors before you begin the procedure.
- Incorrect Driver specification: Ensure that you enter the correct format for the database drivers. For example, with DB2, if you are using the Type 2 JDBC driver, the format should be jdbc:db2:databasename. For Type 4 JDBC drivers, the format should look like the following: jdbc:db2://db2server.domain.com:50000/databasename:returnAlias=0;.

Note: Consult with your database server's documentation to confirm the correct format.

Multiple domains

If the DbUser, DbUrl, and DbPassword properties are not the same values across domains, the *dbdomain.DataSourceName* value should be changed for those domains that differ from the rest.

The value for the *dbdomain.DataSourceName* should not be the same value as *dbdomain.DbName.*

If you are unsuccessful after reviewing your configuration and using various support tools to help you debug, you may need to engage support. Refer to Appendix A, "Using IBM tools to find solutions and promote customer self-help" on page 169 for information about how to prepare your logs before engagement.

3.4.3 LDAP security problem determination

This section shows some common problems with the enable security process, and provides you with some ideas on how to solve them.

Failing to install the required and recommended fixes/Fix Packs for your platform

One of the most common causes of security failures is not meeting the supported hardware or software requirements for your LDAP infrastructure. In addition to meeting the requirements for LDAP, you should ensure that all required and recommended fixes/Fix Packs WebSphere Portal Server have been installed for your platform (refer to 3.1.1, "How do I prepare my system for installation" on page 56).

The fixes/Fix Pack issue is not isolated to the LDAP servers. Not applying the required fixes/Fix Packs for your portal environment can also cause errors during the enablement of security process and can affect the overall operability of your portal environment. To enable security, you should also ensure that you apply the latest WebSphere Member Manager fixes:

`http://www-1.ibm.com/support/docview.wss?rs=688&fdoc=wplcwspm&uid=swg24013740`

Incorrect entries in the wpconfig.properties files

This is perhaps the most common cause of errors with enabling LDAP security. The types of errors are usually attributed to the following:

- Typos or extra spaces: Be certain to look over your properties files for misspellings and extra spaces. Ensure that the values entered are the same case throughout. Running **`validation ldap targets`** before you conduct the enable security task may help you find some of the errors before you begin the procedure.
- Providing incorrect values for LDAP entries: Because the entries in the Advanced LDAP Configuration section are organization specific, **`validation ldap targets`** does not check these entries for errors. Take special care to ensure that the values entered here are correct for your LDAP design, as this is one of the most common causes of failure when enabling security. Verify that you can search for users and groups using the information specified in the Advanced LDAP Configuration using the **`ldapsearch`** utility.

Incorrect privileges for the LDAPBindID

Unless anonymous searches are allowed, the LDAPBindID should have, at a minimum, permission to read and search a subset of the Directory Information Tree specified in the LDAP suffix entry. Confirm the privileges of your LDAPBind user if anonymous access is not allowed.

Failure to disable security before enabling security

Before you can run the enable security task, you must disable WebSphere Application Server Global Security by running the disable security task through the command line or the wizard. If you are encountering problems with running the security tasks, you must disable security each time before you try to reenable security.

4

WebSphere Portal security

IBM WebSphere Portal provides personalized access to applications and processes, ranging from small and simple applications to complex enterprise information systems. It aggregates the content from different data sources to provide a single user interface for centralized display and management. These different applications and systems may require their own security controls with different level of complexities. To accommodate such a wide range of security requirements, WebSphere Portal has provided a rich set of configuration options that integrate with different security infrastructure components for authentication, authorization, single sign-on (SSO), and user management, and the customers can choose the combination that best matches their security needs.

In this chapter, we will cover issues in:

- Planning and designing solutions for portal security
- Configuring and customizing settings in portal security to suit special requirements
- Troubleshooting security related problems encountered during configuration and runtime of a portal deployment

4.1 Planning and considerations

In this section, we will address the basic concepts, planning issues, and considerations while configuring WebSphere Portal security.

4.1.1 The basics

IBM WebSphere Portal provides personalized access to applications and processes, ranging from small and simple applications to complex enterprise information systems. It aggregates the content from many different data sources to provide a single user interface for centralized management. These different applications and systems may require their own security controls with different level of complexities. To accommodate such a wide range of security requirements, WebSphere Portal must integrate with different security infrastructure components for authentication, authorization, single sign-on (SSO), and user management, so that the customers can choose the solution that best suits their security needs.

WebSphere Portal is a J2EE application deployed onto an application server, called WebSphere_Portal within a WebSphere Application Server. It can leverage the underlying application server's powerful security infrastructure. In addition, WebSphere Portal security extended the security configuration provided by the Application Server, and presented a flexible set of options for customers to choose. It also provides the Credential Vault mechanism for supporting Single Sign-On solutions with back-end enterprise systems.

WebSphere Portal security consists of authentication and authorization. Authentication answers question of confidentiality, that is, the user submits the credentials to let the system know who he or she is and the server then verifies whether the user's credentials are correct against a user registry. Authorization is more commonly referred to as Access Control. Once the user's identity is established during the authentication phase, the authorization mechanism of the system checks what the authenticated user can do on which resources on the site. WebSphere Portal utilizes WebSphere Member Manager (WMM) for its user and group management, through an abstraction layer called Portal User Management Architecture (PUMA).

Figure 4-1 on page 87 gives a general overview of the deployment of the WebSphere Portal solution.

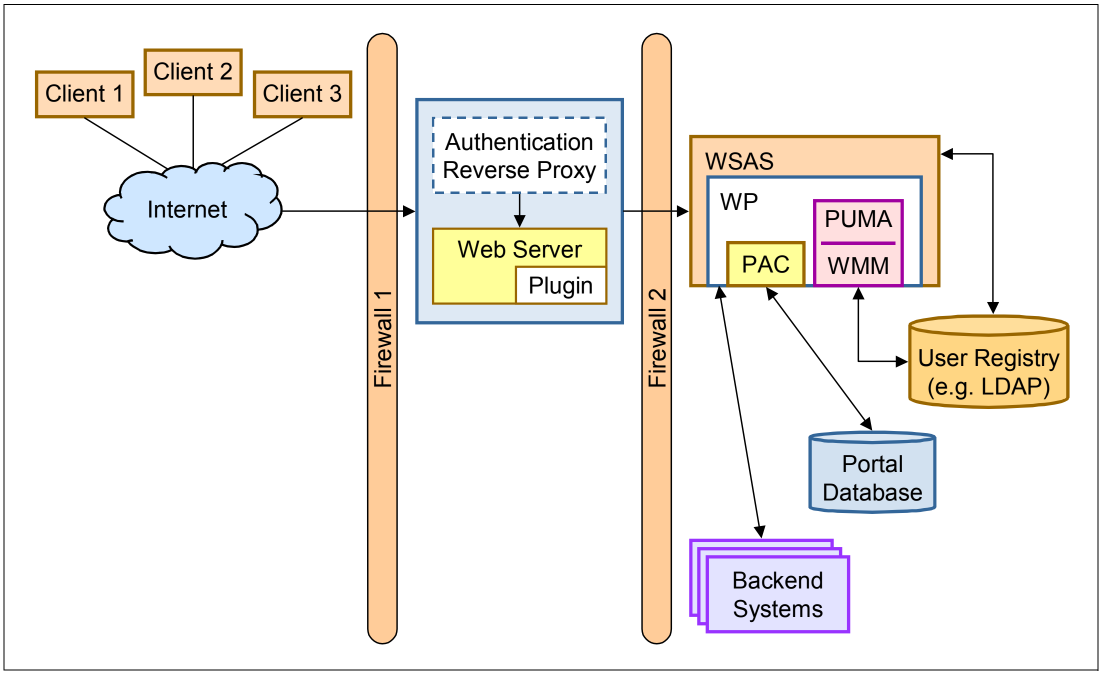


Figure 4-1 The general view of a WebSphere Portal deployment

4.1.2 WebSphere Member Manager (WMM)

WebSphere Member Manager for WebSphere Application Server handles member data and profiles. In the context of WMM, four types of members are supported: Person, Group, Organizational unit, and Organization. Each member has a member profile that describes its characteristics within the system, and differentiates one member from another.

The Member Manager provides the following:

- A common mechanism to access member profiles that are made of attributes regardless of where and how the data of the member profile is stored. These attributes can be single-valued, multi-valued, or composite.
- A set of services to act upon and manage profiles, such as create, read, update, remove, and search members in the profile repository.
- These services also support managing groups, including assigning members to and unassigning members from groups and querying group membership.
- A hierarchical structuring of members.
- A database profile repository adapter to interact with a database profile repository.
- Lightweight Directory Access Protocol (LDAP) profile repository adapters to interact with a set of supported LDAP servers.

In the context of WebSphere Member Manager documentation, the database profile repository adapter is called wmmDB, and the LDAP profile repository is referred to as wmmLDAP, and each adapter is referred to as the wmmLDAP adapter.

Currently, WMM support the following major commercial LDAP servers:

- IBM Tivoli Directory Server
- Microsoft® Active Directory®
- SunOne Directory Server
- IBM Lotus Domino Application Server
- Novell eDirectory

WMM implements the wmmLDAP as an abstraction layer, in which for each type of the supported LDAP servers, WMM provides an adapter module to shield the implementation details of the LDAP servers from application developers. This way, it is able to provide a standard set of Member Repository APIs for applications, like WebSphere Portal, to manage uses and groups.

Optionally, you can use a look-aside profile repository adapter to interact with a look-aside repository using one of the available commercial databases with a schema defined by the Member Manager. The look-aside repository is used to store member attributes that cannot be stored in the member's main profile repository (such as the wmmLDAP). In Member Manager, the look-aside repository is referred to as wmmLookAside and the adapter is referred to as the wmmLookAside adapter. Although you can technically use wmmLookAside in conjunction with wmmDB repository, it is likely unnecessary, since all functionalities supported by the wmmLookAside is also supported by wmmDB.

Every member managed by Member Manager requires a unique identifier. A unique identifier allows a member profile to be easily retrieved. Member Manager provides two types of unique identifiers:

- *memberDN* is a distinguished name for a member, and is convenient for identification and display purposes. memberDN is unique and may be changed and reused. After a member is deleted from Member Manager, a new member can be created and reuse the memberDN of the deleted member. An example of a memberDN of a Person “Jane Doe” is “uid=janedoe,ou=people,ou=sales,o=acme.com”.
- *memberUniqueId* is unique, static, and never reused. That is, once memberUniqueId for a member is created, the value of that memberUniqueId will not be changed, even if the member is deleted. A new member cannot reuse the value of the memberUniqueId of the deleted member.

The memberDN therefore uniquely identifies a member at a single point in time while the memberUniqueId, due to its characteristic of never being reused, uniquely identifies a member over time. In the example above, the person “Jane Doe” may change a job and work for a new organizational unit “marketing”, so the new memberDN then becomes “uid=janedoe,ou=people,ou=marketing,o=acme.com”, but the memberUniqueId is still the same.

The memberUniqueId in WMM can be mapped to a unique attribute in the LDAP server. The examples of memberUniqueId might be *ibm-entryUUID* for IBM Tivoli Directory Server, or *objectGUID* for Microsoft Active Directory.

Depending on your usage of member profile data, you may want to use the memberDN or both the memberDN and the memberUniqueId.

Since memberDN values are readable, they are suitable for display purpose. The memberUniqueId values are not guaranteed to be readable and therefore may be unsuitable for display. Since a memberDN can be changed and reused, if your application receives a memberDN from Member Manager, puts the memberDN in some form of storage, and subsequently uses that memberDN with Member Manager, there is no guarantee that memberDN will not refer to a different member than the one to which it previously referred.

When an application, such as WebSphere Portal, uses Member Manager, the application may have its own application-specific repository for data that is related to the member in Member Manager. This means the application needs a linkage for the data of a member managed by Member Manager and its own application-specific data for the same member. Since the memberDN may be changed and reused, in general it is not suitable to be used as the linkage. However, memberUniqueId, which is unique, static, and never reused, is suitable to be used as the linkage. Still, with the previous example, the memberDN of "Jane Doe" was changed, but her application data should be still linked by the memberUniqueId. Thus, this is why memberUniqueId is recommended to be static and never changed. If this "Jane Doe" leaves the company and a new "Jane Doe" joins the same company, the second "Jane Doe" may have the same memberDN, but she would not be able to access the application data in the system, because the memberUniqueId is different.

In the context of WebSphere Portal, only two member types, Person and Group, are supported. In WebSphere Portal, the member unique identifier (memberUniqueId) is called *external ID*, or *extId*. In Version 6 of WebSphere Portal, Portal Access Control (PAC) utilizes extId as the primary key in permission database tables, linking the users and groups to the access control data.

4.1.3 User registry and member repository

In the context of WebSphere Application Server, a *user registry* stores all user and group data, including the user login ID and password, other user and group attributes, user and group membership information, and so on. In the context of WebSphere Application Server global security, three user registry types are supported. They are the Local Operating System user registry, Lightweight Directory Access Protocol (LDAP) user registry, and custom user registry (CUR).

In some corporations, the existing directory servers, such as LDAP servers, are not capable of handling their needs. For example, a recent merger of two companies cannot consolidate their employees into a single directory in a short period. They may have to run their businesses to accommodate the two coexisting sub-directory server systems. In this case, WebSphere Application Server provides an Application Programming Interface (API) for customers to develop their own custom user registry (CUR). They can also consider other solutions, such as Tivoli Directory Integrator, to provide integration of their multiple directory systems.

Within WebSphere Portal, only LDAP and custom user registries are supported, not the Local Operating System, because of the configuration of the Lightweight Third-Party Authentication (LTPA) mechanism used in Single Sign-On (SSO).

In the context of WebSphere Portal and Member Manager, a *member repository* is the store for user profile data and the group data, and their membership information. Two different terms (user registry and member repository) are used because it is possible for the datastores to be different. For example, when the portal server requires application specific user attributes that are not available in the LDAP server, the administrator can opt to use the LookAside mechanism provided by WebSphere Member Manager. Thus, the member repository has the extension in the LookAside database tables. In most cases, however, the user registry and member repository are in the same datastores.

WMM supports three types of member repositories: database (DB), LDAP, and custom member repository (CMR). In the database member repository (WMMUR DB), WMM had provided its own Custom User Registry (CUR) implementation (using the CUR API provided by WebSphere Application Server) to be used in the application server security configuration.

For accessing the user profile and group information, WMM provides the Custom Member Repository (CMR) module. The two classes are:

- com.ibm.websphere.wmm.registry.WMMUserRegistry (CUR)
- com.ibm.ws.wmm.db.DatabaseRepository (CMR)

and can be respectively found in the WebSphere Application Server security configuration and WMM configuration (wmm.xml).

When a customer user registry (CUR) is developed by the customer, a corresponding custom member repository (CMR) must be coded for configuring WMM. The CMR API is private and unpublished. To obtain this API, IBM support must be contacted and an non-disclosure agreement must be signed.

The security of an out-of-box installation of WebSphere Portal Version 6 is enabled with the WMMUR DB option based on the embedded version of IBM Cloudscape Database. The idea is for the administrator to have a working system right after the installation. This configuration is suitable to a system for a fairly small scale experimental environment, like individual development or unit testing. In general, it is not recommended for a production environment. If the system will be put into production, the future growth and performance impact should warrant a better enterprise solution, based on a commercial database system and a directory server.

Attention: We do not recommend the default out-of-box database configuration in a production environment, mainly for performance and management considerations.

4.1.4 Single sign-on (SSO)

The power of WebSphere Portal is its capability of integrating and aggregating the applications running on the server and other enterprise information systems to present a single user interface. In most cases, the back-end enterprise applications require their own authentication and authorization to the users. Most implementations of the portal solution avoid the requirement of repeating authentication of the users. This is where SSO comes into play.

The goal of single sign-on is to provide a secure method of authenticating a user one time within an environment and using that single authentication (for the duration of the session) as a basis for access to other applications, systems, and networks. In the context of IBM WebSphere Portal, there are two single sign-on realms: the realm from the client to portal and other Web applications and the realm from the portal to the back-end applications.

WebSphere Application Server adopted an authentication mechanism called the Lightweight Third Party Authentication (LTPA) for single sign-on support. The LTPA protocol is intended to work in a distributed environment through cryptography. This support permits LTPA to encrypt, digitally sign, and securely transmit authentication-related data, and later decrypt and verify the signature. The LTPA protocol uses cryptographic keys to encrypt and decrypt user data that passes between servers. These keys must be shared between the different servers. The participant servers of the LTPA SSO must be configured to use the same LDAP or custom user registry. With LTPA enabled, a token is created with the user information and an expiration time, and it is signed by the keys. The LTPA token is time sensitive. All servers that participate in the single sign-on must have their time, date, and time zone synchronized. The LTPA token is passed in the HTTP header as a cookie, called *LtpaToken*. At the receiving end, the token is decrypted to obtain the user information. Due to the nature of sharing keys, it is important to guard the key files and WebSphere Application Server configuration files.

For all details about SSO, LTPA and related topics, refer to the WebSphere Application Server Information Center.

4.1.5 WebSphere Portal login process

It is very important to understand the basic login process in WebSphere Portal security. It is the key in finding the cause of problems in many failure scenarios.

By default, WebSphere Portal security configuration is set up as a form-based login. This is set in the Web application descriptor of the portal servlet and the custom login form is presented by the Login portlet, or the Login screen (for backward compatibility).

The HTTP basic login, as used by many internet Web sites, is not recommended and not supported for WebSphere Portal, because of its limitations.

You can also configure WebSphere Portal to use SSL/TLS client certificates as the login mechanism to provide additional security, or a combination of a form-based login plus the client certificate to achieve a higher level of security.

In this section, we describe the basic login flow in details, and then give a short description of some other possible ways of login. Figure 4-2 shows the general login points to WebSphere Portal.

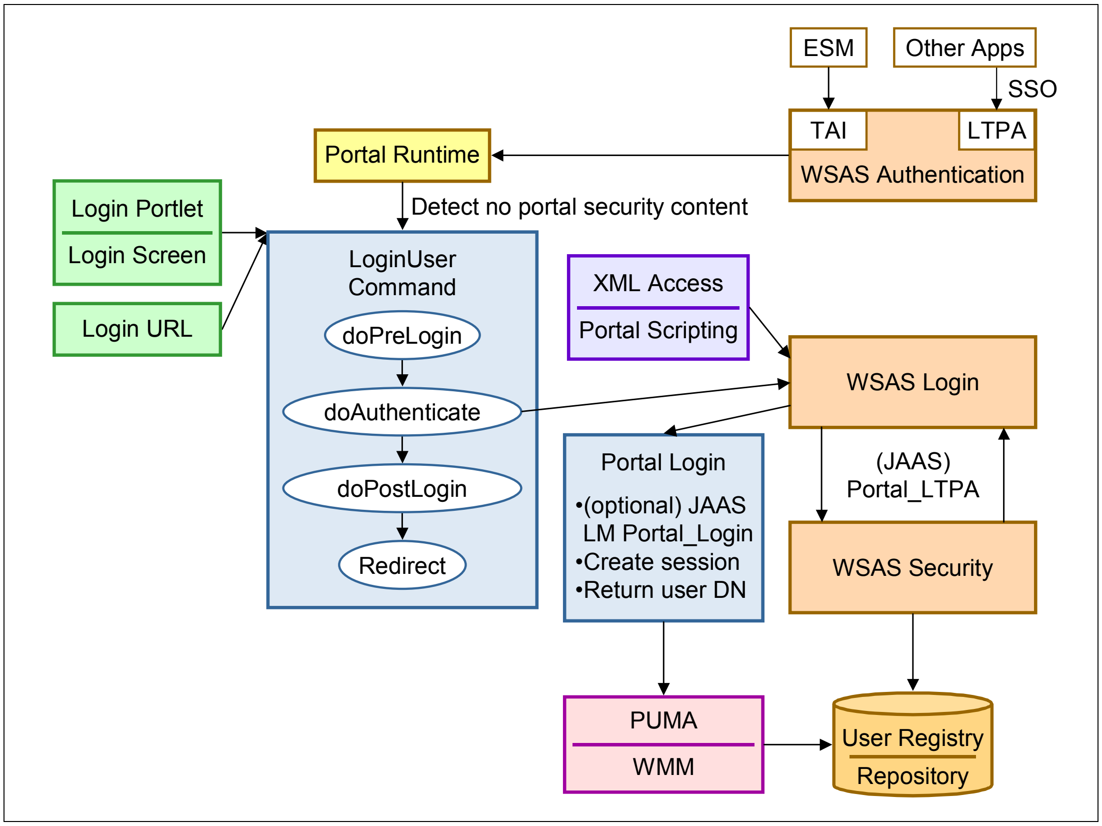


Figure 4-2 Basic Portal Login flow

In its basic login form, after a user submits the user ID and password, the WebSphere Portal code triggers a programmatic Java Authentication and Authorization Service (JAAS) login using a JAAS login module configured in WebSphere Application Server (WSAS), called *Portal_LTPA*. This JAAS login module stack performs the authentication against the user registry configured in WebSphere Application Server, and then, if this authentication succeeds, creates the LTPA cookie. Taking the stackable feature of the JAAS model, the Portal_LTPA JAAS configuration can also be extended by custom login modules, such that administrators and developers can incorporate additional checks specific to their environments.

The result of the login is an object of javax.security.auth.Subject, representing a grouping of related information for a single entity, such as a person. It contains identities (represented as a javax.security.Principal object), and security-related attributes (such as passwords or cryptographic keys) as credentials. This “WebSphere Application Server subject” forms the security context associated with the authenticated user and can be used by the Application Server login to pass the context to the WebSphere Portal Login. WebSphere Portal then can retrieve the Application Server user identity to perform additional portal related login steps.

WebSphere Portal Login first creates the “Portal subject”, which is represented by a javax.security.auth.Subject object. This second subject represents the same unique security name and LTPA token as the WebSphere Application Server subject and is populated with the same principals and public credentials. The difference in this Portal subject is that, unlike the WebSphere Application Server subject, it is not locked down after the successful authentication; it can be modified later on by any portlet using the Credential Vault portlet service. Another difference is that the Portal subject is not shared with applications besides WebSphere Portal.

The Portal subject is also passed on to the optional Portal JAAS login. Depending on the configuration, WebSphere Portal triggers a second JAAS login called Portal_Login, which you can extend using custom login modules.

WebSphere Portal Login also creates the session unless the session has already been created for the anonymous user (when “public.session” is set to true in portal ConfigService), or there is no need for a Web Container session (if the XMLaccess command interface or the portal scripting interface triggered the login process). Finally, the portal user object is returned to the calling code.

There are many ways to trigger the user login process. They are summarized in Table 4-1.

Table 4-1 Login access points

Login point	The flow
Normal portal login using the Login Portlet or Login Screen.	Triggers the engine command LoginUser, which essentially triggers the Login process described in this section.
Direct access to a protected URL.	The application server redirects to a redirect form, which invokes a special servlet filter for detecting virtual portal login, and then redirects to the virtual portal login form.
Portal direct login URL.	The Portal login URL directly executes LoginUser engine command through the encoded URL.
XMLaccess command-line interface	The XMLaccess servlet called by the client directly triggers the application server login using the user credentials taken from the command arguments. When XMLaccess is triggered within the Portal administration GUI, the security context is the current user credentials.
Portal Scripting Interface	Triggers the WebSphere Application Server and Portal Login by the Portal Scripting MBean. The user credentials are taken from the $Login scripting command.

Login point	The flow
Other applications through SSO.	The LTPA in the client request triggers WebSphere Application Server to create the security context with the user credentials and passes it to Portal login. The LoginUser engine command is then triggered.
External Security Manager through TAI	WebSphere Application Server checks LTPA first. If none exists, the TAI configured in WebSphere Application Server trusts the ESM and creates the LTPA for the user, and sends the security context to Portal Login.
SSL client certificates	WebSphere Application Server authenticates the user through the client certificate, and builds and passes the subject to Portal Login

To obtain details, refer to the white paper *Understanding and configuring WebSphere Portal login and logout*, found at:

http://www.ibm.com/developerworks/websphere/library/techarticles/0706_buchwald/0706_buchwald.html

4.1.6 Portal Access Control (PAC)

The access level of a user to a portal resource is measured by the actions he can apply on the resource. In the portal environment, these actions are view/read, update/write, delegate, traverse, and delete. For different types of resources, different set of actions are applicable. A set of fixed role types are defined in portal access model for management, each one of which is represented by a set of actions (called actionset) that can be applied to the resources.

The Portal Access Control (PAC) authorization model is based on the concepts of protected resources and the hierarchy these resources build up. It contains a set of fine-grained configurations for portal resources, such as pages, portlets, services, and global settings. They provide a full range of control settings from an easy and simple solution to fairly complicated enterprise level systems.

The artifacts defined by the PAC model are summarized in Table 4-2.

Table 4-2 PAC artifacts

Artifact	Description
Protected Resources	Represent a set of portal artifacts protected by the portal, and they are divided into four domains.
Protected resource hierarchy	Starting from a set of virtual resources to form a tree structure, with virtual resource PORTAL at the top root.
Virtual resources	A set of virtual objects created during portal installation to form the roots of the protected resource hierarchy.
Role types	Formed by the action-sets that can be applied to resources.
Role	An instance of a role type with a specific resource.
Role block	A configuration set to block role inheritance or propagation.
Ownership	Unrestricted access to the resource by the owner.

Understanding the hierarchy of protected resources is the key to having a clear picture of the permissions assigned to the nodes on the tree. The permission inheritance plays a crucial role in the runtime decision making of the portal access control. Figure 4-3 shows the tree of protected resources within WebSphere Portal.

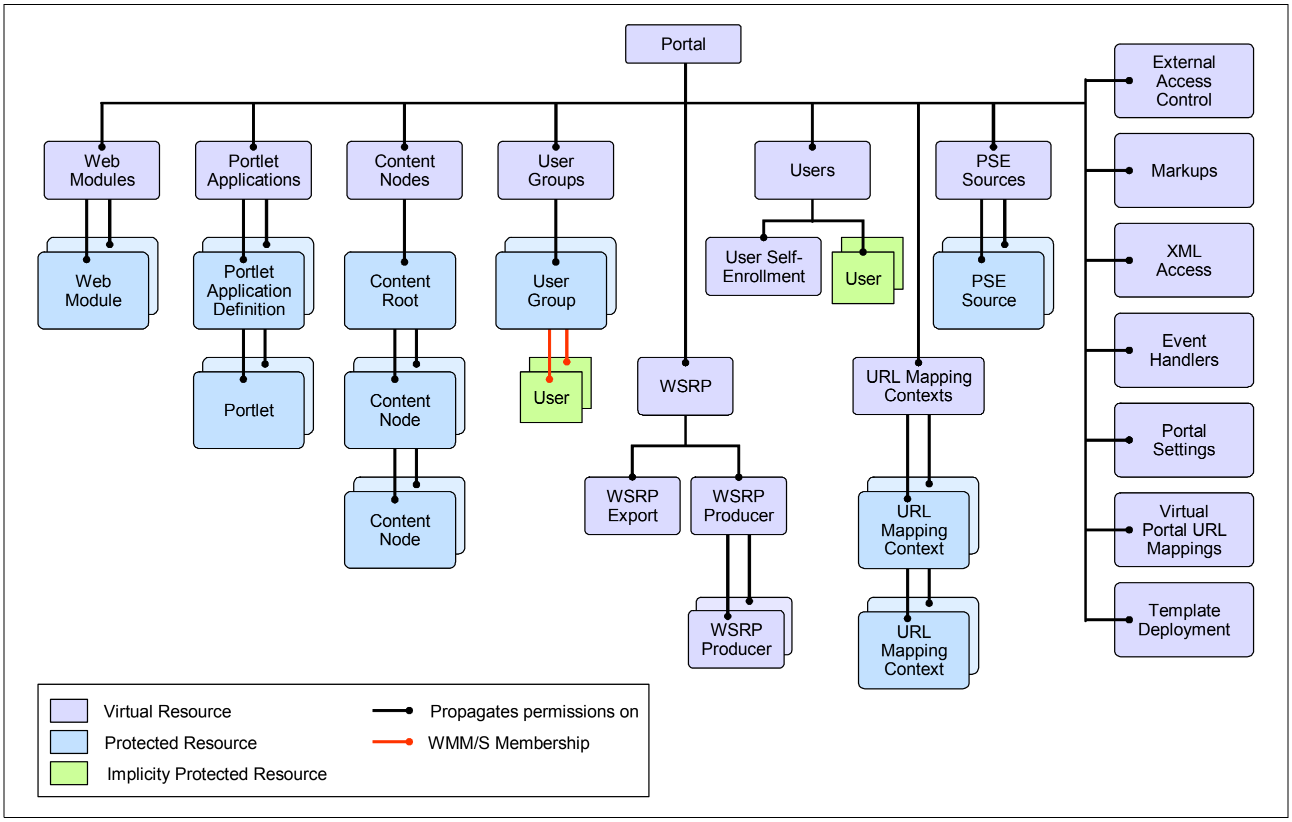


Figure 4-3 The tree of WebSphere Portal protected resources

PAC is the single decision point within the WebSphere Portal. It controls the access to all protected portal resources. Figure 4-4 on page 95 showed the basic components of PAC. The central piece of PAC is the Access Control Engine that implements the PAC API and provides the core support functions to different components:

- The dynamic permission configuration is accomplished through one of the three ways: the admin portlets, Resource Permission Portlet, and User and Group Permissions Portlet, the configuration utility called XMLaccess, or the Portal Scripting Interface (**`wpscript`**). They directly call a set of Access Control commands that in turn call the AccessControlConfigService.
- The portal runtime decision module is triggered when a resource is accessed by a user. Most of the permission configurations should be assigned to groups, which is more efficient than assigning them to individual users. Thus, one should carefully design the LDAP group structure and user membership assignment. WebSphere Member Manager Portal supports different group structures: static, dynamic, mixed, and nested groups. Portal runtime access decision are made by calling AccessControlService.
- When WebSphere Portal is configured to use an external authorization engine, such as the Tivoli Access Control authorization server, portal provides a set of Service Provider Interfaces (SPIs) that can directly interact with Portal Access Control Engine by calling ExternalAccessControlSerivce.

When the PAC configuration is to be persisted, the datastore persistence layer is called to pass the configuration data to the portal database. The Portal Access Control runtime decision module has to retrieve the persisted permission data through the datastore persistence layer. In order to reduce the IO traffic to the datastore, the portal architecture adopts a fairly sophisticated cache management system.

The multi-level fine-grained controls over PAC caches gives the portal administrator a lot of flexibility and opportunities in tuning the performance. On the other hand, improper PAC cache settings could adversely cause serious performance degradation to the portal system.

In WebSphere Portal Version 6, the PAC and other cache configurations are managed by CacheManagerService. A default set of configuration parameters are presented in CacheManagerService.properties in <portal_root>/shared/app/wp.services.properties.jar. These settings can be customized through "WP CacheManagerService" in the WebSphere Administrative Console by selecting **Resources** → **Resource Environment Providers** → **WP CacheManagerService**.

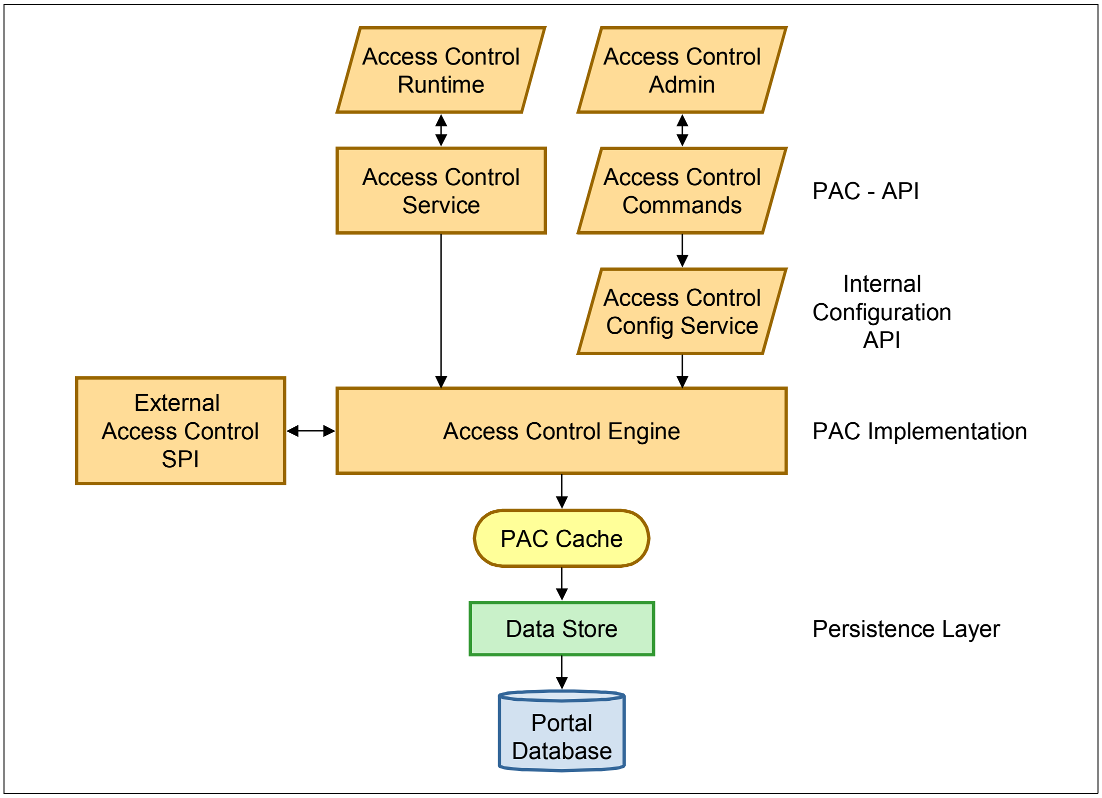


Figure 4-4 Portal Access Control components

Some suggestions on tuning the PAC caches are summarized below:

- Keep the access control configurations as simple as possible.
- Minimize the number of user groups.
- Minimize the number of different groups to which the users belong.
- Avoid nested group hierarchies and depth of the nested groups.
- Avoid doing access control administration while the system is under heavy load.
- Limit the use of external access control.

The general guidelines for configuring PAC are summarized in the white paper *Performance tuning of Portal Access Control*, found at:

http://www.ibm.com/developerworks/websphere/library/techarticles/0508_buehler/0508_buehler.html

Although this white paper was written for Version 5, many principles are still applicable to Version 6.

4.1.7 Secure communications over SSL

Secure communication over the wide-open unprotected internet is essential to many business applications. It builds up consumer confidence and protects sensitive data transmitted through the internet. Secure Socket Layer (SSL) or its successor Transport Layer Security (TLS) are the protocols that leverage a variety of cryptographic algorithms to implement security.

Even within corporations, the communication through the intranet is not necessarily safe. As a matter of fact, reported internal attacks constitute an alarming 45 - 50% of the total cases. Sensitive information passed around the corporate networks are subject to attacks by disgruntled or dishonest employees. Companies put themselves at risk by holding and passing sensitive information without protection. Internal threats can generally be categorized as the following three types:

- Corporate espionage: Employees or contractors may be recruited and paid by competitors to steal company secrets.
- Malicious employees: Current and recently terminated employees may want to cause damages to the company by destroying valuable data or files, or causing network disruption.
- Unintentional breaches: Employees put the network at risk by installing unauthorized software, opening virus-infected e-mail attachments, succumbing to social network attacks, and so on.

When designing your Web sites based on WebSphere Portal, you should understand clearly what data is sensitive and needs protection. Depending on the nature of the application, you may want to secure the entire site or only a portion of it. The WebSphere Portal infrastructure provides the flexibility of a range of solutions that suit your requirements.

On the other hand, you have to understand that there are performance implications when configuring SSL due to its protocol nature, that is, a handshake phase is required to establish the trust relationship between the communication parties, and then there is an exchange of keys. In addition, all communications over SSL channels must be encrypted at the source and decrypted at the destination. This process will impact processing on all requests going through the secured channel. Also, the configuration makes certain cache options impossible. Depending on the encryption algorithm, the length of the encryption key, the complexity of the data, and other factors of the network, the overhead of SSL can be between 10 - 50%. In most cases, using SSL accelerator will help performance.

4.1.8 Integration with Tivoli Access Manager and WebSEAL

For WebSphere Portal authentication, you can use the native authentication mechanism provided by the underlying WebSphere Application Server infrastructure, or an external security manager such as Tivoli Access Manager for e-business (TAM). The integration of WebSphere Portal and TAM provides a single central authentication point for one or more systems and other Web applications, thus providing easier management of security assets.

WebSEAL, a component in Tivoli Access Manager, acts as a reverse proxy server that intercepts all Web requests coming into the portal Web site. When a protected resource is accessed and the user has not been authenticated, WebSEAL challenges the user by consulting with its authorization server (policy server) and the user registry, so the reverse proxy is able to verify the user's identity and pass the user's identity info through iv-user and iv-creds in the HTTP header to WebSphere Application Server. The application server's Trust Association Interceptor then trusts the authentication by retrieving and verifying the user information from the same user registry, and then creates an LTPA token for single sign-on.

Two important considerations:

- Before you can configure the integration with TAM, the security of WebSphere Portal must be configured and single sign-on must be working correctly with the LTPA token.
- The same user registry must be configured for WebSphere Application Server and TAM.

4.2 Security configurations and customizations

Here we discuss some basic configurations and customization scenarios.

4.2.1 The default security configuration

WebSphere Portal Version 6 is installed with the security enabled against WMM UR with the WMM database as the user registry. With this setting, the portal is functional right after installation and this configuration is suitable for a simple environment, like unit testing or development.

As we specified in 4.1, “Planning and considerations” on page 86, the administrators should seriously consider reconfiguring security with a commercially available LDAP server. If the system will be put into production and performance is a major concern, we do not recommend the database solution and you should reconfigure security as early as possible, so that the impact of the user registry migration can be kept to a minimum.

The portal configuration tasks are designed to configure portal security to a default configuration according to the entries in the file wpconfig.properties. It does not deal with more specific customized environments. In these cases, manual modification to the configuration files may be required. The configuration changes we are dealing within these cases are mainly within WMM files and WebSphere Administration Console.

WMM requires an external identifier (extId) to be defined and mapped to a unique LDAP server attribute. WMM supports four different member types: Person, Group, Organizational Unit, and Organization. Thus, this external identifier must be mapped to an attribute that all four member types have; otherwise, WMM will fail. Since WebSphere Portal only uses two member types, Person and Group, in most cases, you can disable the support for OrganizationalUnit (OU) and Organization (O) in wmm.xml, wmmLDAPServerAttributes.xml, and wmmAttributes.xml, by simply eliminating the tags or parameters that are related to OU and O.

Extending or restricting user populations to include or exclude LDAP branches can be done by adding or deleting search bases, broadening or narrowing the search bases, or adding search filters within the WMM configuration. When you extend the user search bases, you need to be cautious that you do not give access to your portal to some users that are not supposed to have.access to the portal.

4.2.2 Reconfigure security

In WebSphere Portal Version 6, the resource permissions are all keyed on the extId of the users or groups. This makes the security reconfiguration much more involved. The reason is that switching the LDAP server implies all extIds used for resource permissions will be invalid in the new LDAP server, if extIds are mapped to a unique attributes created by the LDAP system, such as ibm-entryUUID (IBM Tivoli Directory Server), or objectGUID (Microsoft Active Directory). Thus, the simple procedure of “disable-reenable” security may wipe out all of the Portal Access Control configuration. The solution is to have a full XMLaccess export on all protected resources in all domains. and a full XMLaccess of the users and groups before security reconfiguration, and import these XML files back to recreate the permission configuration. If Application Groups are used, they must be recreated by importing the users and groups XML file before the permissions can be recreated.

Important: Do not run “disable-security” before you understand its consequences. If you are unsure, contact IBM WebSphere Portal support before taking any action.

4.2.3 Change user IDs and passwords

For portal security configuration, there are mainly four user IDs and one group required in the LDAP. To successfully run the configuration task, however, you also need a couple of groups for WebSphere Web Content Management and Portal Document Manager. In this Redpaper, we only discuss the issues with Portal configuration and problem determination, and the users and groups used in Portal and WebSphere Application Server.

The four user IDs and one group are referenced in wpconfig.properties as:

- WasUserid: This is the administrator user for WebSphere Application Server, sometimes called Server ID. You use this ID to start and stop the server, and to log on to the administrative console for any administration configuration on the application server. This user ID can be any user in the LDAP server. It does not necessarily have any rights in the LDAP.
- LDAPBindID: This is the user ID that WebSphere Application Server uses to bind to the LDAP server. It must be able to authenticate user IDs and have the necessary access rights (read/write/modify/delete) on the LDAP server, depending on how the application server is configured to use the LDAP server.
- PortalAdminId: This is the portal administrator user. It is the most important user in portal configuration, but this user can be any LDAP user that can be searched. Make sure you always specify the full user Distinguished Name (DN) on this line.
- PortalAdminGroupId: This is the portal administrator group. Any user IDs in this group should have the same administrative rights as the portal administrator user does. In some cases, you can disable the portal administrator user ID and only administrate your portal server using user IDs within this group
- LDAPAdminUId: This user ID is used by WebSphere Member Manager to bind to the LDAP server for its inquires to the LDAP. Like LDAPBindID, it should have the necessary access rights to be able to operate on the sub trees in the LDAP such that portal can make changes to the users and groups.

Since these user IDs can be all different at one extreme or all the same at the other, when you make any changes to the users, you have to understand the implications.

In the following discussion, we assume the user IDs used for the purposes above are all different. After the discussion, readers can easily extrapolate the cases if the user IDs may play multiple roles.

The portal Admin user's password is not stored in any of the portal databases, unless the security is enabled using the database as the user registry, such as the default WMMUR DB. So the password of the portal admin user can be changed through the portal Edit My Profile page, if portal is configured to be able to do so, or can be changed directly in the LDAP server. After the password change, the portal admin user should work fine, but you may find exceptions during the portal startup. This is due to RunAs roles configured on some of the enterprise applications deployed on the Application Server. Check the ones listed here:

- LWP_CAI
- LWP_Security_ext
- LWP_TAI
- pznscheduler.ear

> The portal admin user ID and the group DN cannot be simply replaced without re-configuring security, which mainly involves disabling security, modifying LDAP information in wpconfig.properties, and re-enabling security.

The WebSphere Application Server admin user can be a little trickier, since the password is stored in configuration XML files. Timing is the key. The password should be updated in the Administrative Console. Before the password is changed in LDAP, you must have the Application Server running and already logged in to the Administrative Console. After the password is changed on the LDAP server, you can then change the password in the admin console. Restart the server to make sure the change is successful. Within a cluster, the password should be changed through the Deployment Manager.

The process of changing the password of LDAPBindID is similar to that of the WebSphere Application Server admin user.

The password for the WMM bind user ID (LDAPAdminUId) must be encrypted by using wmm_encrypt.bat/.sh, and written into wmm.xml (adminPassword).

4.2.4 Adding application specific attributes to users and groups

With an LDAP server configuration, a set of default attributes have already been defined based on a standard objectclass, such as inetOrgPerson for users. In many cases, some new attributes, not available in the standard objectclass, are required for the applications. There are a couple of ways to accomplish this task.

You can extend an existing standard LDAP objectclass such as inetOrgPerson to incorporate the new attributes. This must be done using the LDAP server utility and in the LDAP server. In the WebSphere Member Manager (WMM), you need to add this new objectclass for read or write objectclasses in wmm.xml. For example, assume the new objectclass you defined is called acmePerson. This objectclass should be added in wmm.xml, as shown in Example 4-1.

Example 4-1 Customized objectclass acmePerson added in wmm.xml

```
<supportedLdapEntryType name="Person"
               rdnAttrTypes="uid"
               objectClassesForRead="inetOrgPerson;acmePerson"
               objectClassesForWrite="inetOrgPerson;acmePerson"
               searchBases="ou=people,ou=dept,o=acme.com"/>
```

The attributes introduced in this customized objectclass should be added to both wmmAttributes.xml and wmmLDAPServerAttributes.xml.

You can also use the LookAside repository provided by WMM, with the understanding that the LDAP server is read-only or that extending an objectclass is not feasible. To enable LookAside, we recommend that you set “LookAside” to true in wpconfig.properties when enabling security configuration. We also recommend that you add the new attributes into wmmLAAttributes.xml and wmmAttributes.xml before running the security configuration task. If you are not able to decide what attributes to add before enabling security, then you can add the attributes to LookAside DB tables later using the utility “attributeloader” provided by WMM. The process was documented in TechNote 1225316, which can be searched for at:

http://www-306.ibm.com/software/genservers/portal/support/

4.2.5 Integration with Tivoli Access Manager (TAM)

The most common configuration of the integration is for the portal to take advantage of TAM’s centralized security infrastructure, use WebSEAL as its reverse proxy, and leverage the Trust Association mechanism provided by the WebSphere Application Server. WebSphere Portal has designed a set of configuration tasks to configure portal servers for authentication, authorization, and vault adapter.

In order to integrate WebSphere Portal with Tivoli Access Manager and WebSEAL, you must first configure the portal security with native WebSphere Application Server, and verify that it is working correctly with its single sign-on mechanism.

> **Important:** WebSphere Portal security must be configured and tested correctly before configuring TAM or any other external security managers.

The portal configuration tasks for TAM integration are enable-tam-all, enable-tam-tai, enable-tam-authorization, and action-esm-tam-update-vaultservice. enable-tam-all is simply a combination of the other three sub-tasks. These tasks are designed to work under general configurations, and to provide a convenient interface for customers to use. If special treatments are required, manual steps should be taken after running them.

Before the portal server can talk to the TAM Java Runtime (AMJRTE), certain conditions must be set by the configuration task run-svrssl-config, which runs two PDadmin utilities PDJrteCfg and SvrSslCfg sequentially. This task creates a user account and server entries that represent the WebSphere Portal, and in addition, the file PdPerm.properties and a Java key store file are created locally under the Java runtime directory on the portal server box. This client certificate permits portal server to use TAM authentication services. The default expiration date of this client certificate is 365 days.

> **Important:** If the TAM runtime is not configured before, run-svrssl-config should be run first to set up the environment.

> **Important:** Update the client certificate before it expires. Otherwise, it may bring the entire site down.

The portal configuration tasks cannot be used to reconfigure the client certificate. You have to run the following commands from the PDadmin command line:

```
# unconfig
java.com.tivoli.pd.jcfg.SvrSslCfg -action unconfig   \
  -admin_id sec_master -admin_pwd <password>  \
  -appsvr_id <pdservername> \
  -policysvr policyserver.acme.com:7135:1   \
  -cfg_file <java_home>/jre/PdPerm.properties
```

and

```
#  config
java.com.tivoli.pd.jcfg.SvrSslCfg -action config    \
  -admin_id sec_master -admin_pwd <password>  \
  -appsvr_id <pdservername> -port 7223   \
  -policysvr policyserver.acme.com:7135:1   \
  -authzsvr authzserver.acme.com:7136:1   \
  -cfg_file <java_home>/jre/PdPerm.properties  \
  -key_file <java_home>/jre/pdperm.ks   \
  -cfg_action replace
```

where <pdservername> is the server host name you used to run SvrSslCfg to register with the TAM Policy Server, <java_home> is where Java is installed under WebSphere Application Server, and "authzserver" is the TAM Authorization server.

It is crucial to make sure the entries you entered into wpconfig.properties are correct. The configuration tasks in WebSphere Portal take the values of the parameters in the file to assemble and issue PDadmin commands based on the parameters to create the corresponding TAM components.

- enable-tam-tai: This task does three things:
  - Takes the parameters in wpconfig.properties and creates the WebSEAL TAI junction.
  - Configures the WebSEAL TAI in WebSphere Application Server and enables it.
  - Updates "WP ConfigService" to add timeout.resume.session and set it to true.
- enable-tam-authorization: This task consists of the following sub-tasks:
  - Creates the TAM JAAS Login Modules WSLoginModule and PDLoginModule.
  - Creates the property file "callbackheaderslist.properties" with iv-user and iv-creds.
  - Updates "WP ExternalAccessControlService" to set up properties for WebSphere Portal to communicate with the TAM Policy Server.
  - Updates "WP AccessControlDataManagementService" to set the external cache timeout to 300 and whether the roles are reordered for easier reading.
  - Updates "WP AccessControlService" to enable Externalization.

  - Updates “WP AuthencationService” to enable the JAAS login module Portal_Login.

  As of the writing of this Redpaper, portal development is testing a new configuration task for supporting TAI++, with which we no longer create callbackheaderslist.properties and the requirement of the JAAS Login module Portal_Login. Check the portal support Web site for the APAR.

- action-esm-tam-update-vaultservice: WebSphere Portal comes with a default vault adapter for storing the credential vaults used in portal applications. The vaults are stored in the portal database. Alternatively, you can configure TAM’s Global Sign On (GSO) lockbox to store the credential vaults. That is when you need to configure TAM vault adapter, which is done by running **`action-esm-tam-update-vaultservice`**. This task basically takes the parameters and sets up the four custom properties in WP VaultService:
  - vault.AccessManager.vaultadapter=com.ibm.wps.services.credentialvault.AccessManager41VaultAdapter
  - vault.AccessManager.config=accessmanagervault.properties
  - vault.AccessManager.manageresources=true
  - vault.AccessManager.readonly=false

Customizations

The configuration tasks are limited to general configurations applicable to most customer scenarios. If the steps documented in WebSphere Portal infoCenter are followed, you should have a working system after running the tasks. If there are special customizations required on the junctions created from the TAM side, or special requirements on the TAI from the WebSphere side (for example, TAI++), manual steps are required.

If you are configuring an LTPA junction on WebSEAL, you should not configure TAI on WebSphere Application Server. That means you should not run any of the configuration tasks above. Instead, you should create the junction through the TAM PD admin interface to the HTTP server. You should make sure the LTPA key is generated from the WebSphere Application Server and shared among the SSO participating servers.

With the LTPA junction, when the requests are passed to WebSphere Application Server, the LTPA is already associated with the requests, so WebSphere Application Server would treat the requests as being authenticated. It would then retrieve the user info from the token and build up the security context.

In order to configure TAI++ to take advantage of this new WebSphere feature, manual steps are required as of the writing of this Redpaper. Refer to WebSphere Application Server InfoCenter for details

4.3 Problem determination

In this section, we are not going to discuss the step-by-step process of debugging different scenarios. There are millions of reasons something can go wrong. Here we only present some general principles and guidelines to help users of WebSphere Portal to understand the general procedures in troubleshooting their problems.

4.3.1 General problem determination recommendations

Here we discuss some general problem determination recommendations.

Document system changes

You should always document the system changes made, no matter whether it is a configuration change, or deployment of applications, or a Fix Pack or interim fixes. The change logs should be made available online, such that other people have access to them later even after you have left the project.

The change journal or log can be as simple as the ones shown in Table 4-3.

Table 4-3 Configuration change log

Date	User ID	What
Apr 5, 2007	wpsadmin	Transferred database from Cloudscape to DB2.
Apr 20, 2007	janedoe	Installed Employee portlet application.
Aug 7, 2007	wpsadmin	Reconfigured security.
Sep 9, 2007	wpsadmin	Ran XMLaccess import to fix page order.

You can add more information in the “What” column if you wish. Always make a backup copy of the files you are going to change and save them to a separate location or a different hard drive. The change log and these backup files should provide sufficient knowledge to recover the system in case something goes wrong.

Do not to make multiple major changes at the same time. For example, do not configure HTTP over SSL and TAM integration at the same time.

Before making any major changes, such as installing or upgrading the system or configuration changes, you should always back up the system, including the database, LDAP, and the file system. You should try to make these backups approximately at the same time, if possible. See Appendix B, “Maintenance: Fix strategy, backup strategy, and migration strategy” on page 207 for details.

Set up a security audit on the system

We highly recommend the AuditService be enabled all the time on all system environments. For user and group management and portal access control purposes, we suggest the events list shown in Table 4-4.

Table 4-4 audit log

Event name	What is logged
audit.groupEvents	Group creation, modification, and deletion
audit.userEvents	User creation, modification, and deletion
audit.ownerEvents	Owner change of a resource
audit.resourceEvents	Resource creation, modification, and deletion
audit.userInGroupEvents	addition of a user to a group

The audit log entries would look like the ones shown in Example 4-2.

Example 4-2 Audit log examples

```
[08/08/07 19:07:37:703 EDT] I Audit
0000011447bbb24a000000020000069d84c32de6073235ad5834768ac19ebc8ad33e21210000011447
bbb24a000000020000069d84c32de6073235ad5834768ac19ebc8ad33e212100000001 EJPSN0014I:
User [uid=wpsadmin,ou=people,ou=dept,o=acme.com] has created a Resource for
ObjectID = 6_AoJ234SKG10GA5C0252FNRKA0000 and Name = Audittest
[08/08/07 19:08:40:953 EDT] I Audit <no transaction> EJPSN0004I: User
[uid=wpsadmin,ou=people,ou=dept,o=acme.com] has created a User with ID =
uid=audittest,ou=people,ou=dept,o=acme.com, Name = audittest and ObjectID =
9eAe1JC03JH6GP00JMGC13DEJM46GHC4MM47KHC6JMCC4JCCMG16M9023OCCM1
```

You can see that the time stamp, the user ID, the resource name, and ObjectID are all clearly logged in the file. These audit logs should be saved for the lifetime of the portal system. Using a custom script, you may be able to generate a change log from the audit log files.

Understand and recreate the problem

When a problem is reported or encountered, information about the problem should be collected as soon as possible while the memory is fresh. A clear description of the problem and how to recreate it should be documented.

The following list of questions may help you better understand the situation and collect as much information as possible:

- What is the problem? How can you describe the problem?
- Are there any error messages? Is a screen capture available?
- When did it happen? Under what conditions was the problem observed? Can this problem be easily recreated?
- Have there been any changes done on the system components? If yes, what kind of changes have been made in the past two weeks before the problem started?
- Does this problem affect all users, including the portal administrator user? If only some users are affected, what are the major differences about the affected users and those who are not affected?
- Are there any special conditions that were met when the problem started? How can you tell whether your recreation of the problem is the same?

We should try to simplify the situation and eliminate the factors to isolate the problem. If you are not sure whether some components or users have anything to do with the problem, for example, try to remove them from the situation and see whether you can still recreate the problem.

If the problem can be recreated, clearly document the steps of recreation, the conditions under which the problem had happened, and the user or group information that can be used to recreate it.

When recreating a problem, the time should be as accurate as possible. If a specific user is required, get the user full DN and an LDIF output of the attributes and the groups the user belongs to.

Save all the relevant documents for analyzing security related problems, and collect WMM configuration files in <wp_root>/wmm directory, wpconfig.properties, security.xml, and log files, such as ConfigTrace.log, SystemOut.log, and SystemErr.log, as well as trace.log, if any traces are enabled. Always keep the evidence for the “crime scene”.

A verification checklist of a working system with security enabled

After the security is enabled, the first thing we would like to do is to verify whether the configuration is correct. The following is a generic list, which should be applicable to most cases:

- The configuration task ran successfully. (BUILD SUCCESSFUL!)
- You can start the application server and portal server using the administration user (WasUserid) configured for WebSphere Application Server.
- You can log in to WebSphere Administrative Console and Portal using the administration users respectively.
- The portal administration users are able to navigate to the administration portlets and conduct administration operations, such as create pages, search and add users and groups, install portlets, create virtual portals, and so on.
- The portal administrator user can assign users and groups of access permissions to resources, and verify them with proper user IDs

4.3.2 Typical portal traces for different security scenarios

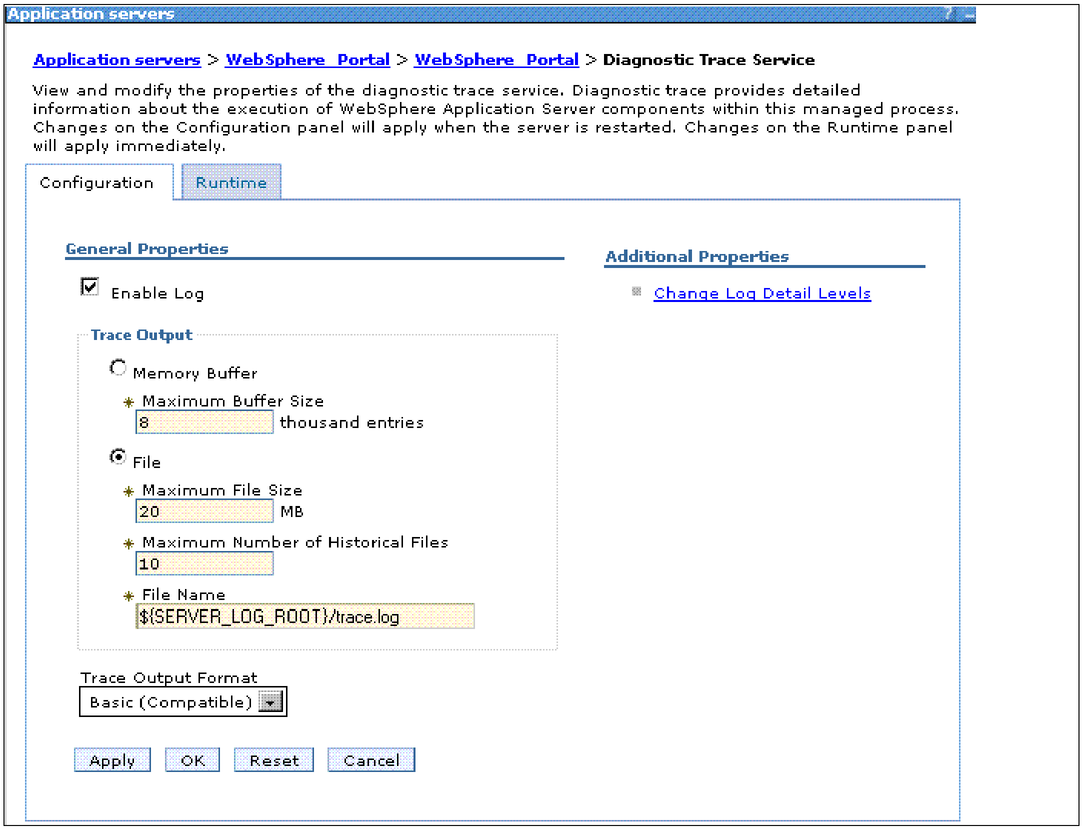
Here we discuss typical portal traces for different security scenarios.

Recommendations for log settings

How many times have we seen that customers spent several hours or even days to try to recreate a scenario, and ended up with a small log file without capturing the real data? This happens often because the default settings for runtime log and diagnostic trace files are usually too small, and should be changed to accommodate more data.

> **Important:** Always change the default size and historical copy settings for the runtime and diagnostic trace log files, such that critical error conditions are logged.

Depending on the environment, you may want to increase the size of log file and the historical copies of these files to a larger value. For example, you can set the file size to 20 MB and the number of files to 10. Thus, you would effectively have about around 200 MB of log data at any time for analysis. Figure 4-5 shows the example settings.



Application servers

Application servers > WebSphere_Portal > WebSphere_Portal > Diagnostic Trace Service

View and modify the properties of the diagnostic trace service. Diagnostic trace provides detailed information about the execution of WebSphere Application Server components within this managed process. Changes on the Configuration panel will apply when the server is restarted. Changes on the Runtime panel will apply immediately.

Configuration | Runtime

General Properties

☑ Enable Log

Trace Output

○ Memory Buffer

* Maximum Buffer Size: 8 thousand entries

● File

* Maximum File Size: 20 MB

* Maximum Number of Historical Files: 10

* File Name: ${SERVER_LOG_ROOT}/trace.log

Trace Output Format: Basic (Compatible)

Apply | OK | Reset | Cancel

Additional Properties

- Change Log Detail Levels

Figure 4-5 Trace log settings

> **Tip:** In a stand-alone environment of WebSphere Portal Version 6, the WebSphere Application Server Administrative Console can be accessed through the application WebSphere_Portal (the default port 10027). You do not need to start server1.

Portal traces

After the initial troubleshooting steps, if it is not obvious what had caused the problem, you may need to enable additional traces. Here we try to give some generic trace strings that are applicable to many situations. If you have reason to believe that more specific strings are required, we would suggest an analysis of the Java stacktrace following the error message(s) in the log. The stacktrace should show certain calling code patterns that should give clues to what to trace.

On the WebSphere Portal support Web site, we have published a set of MustGather TechNotes that contain some typical cases for security related problems, covering a broad range of problems encountered in debugging portal security problem. They are accessible at:

`http://www-1.ibm.com/support/docview.wss?rs=688&uid=swg21236371`

For most Portal security related problems, we recommend WMM traces (<*wmmbase*>):

```
com.ibm.websphere.wmm.*=all:com.ibm.ws.wmm.*=all:WSMM=all
```

The most commonly used Portal security trace strings (<*authbase*>) are:

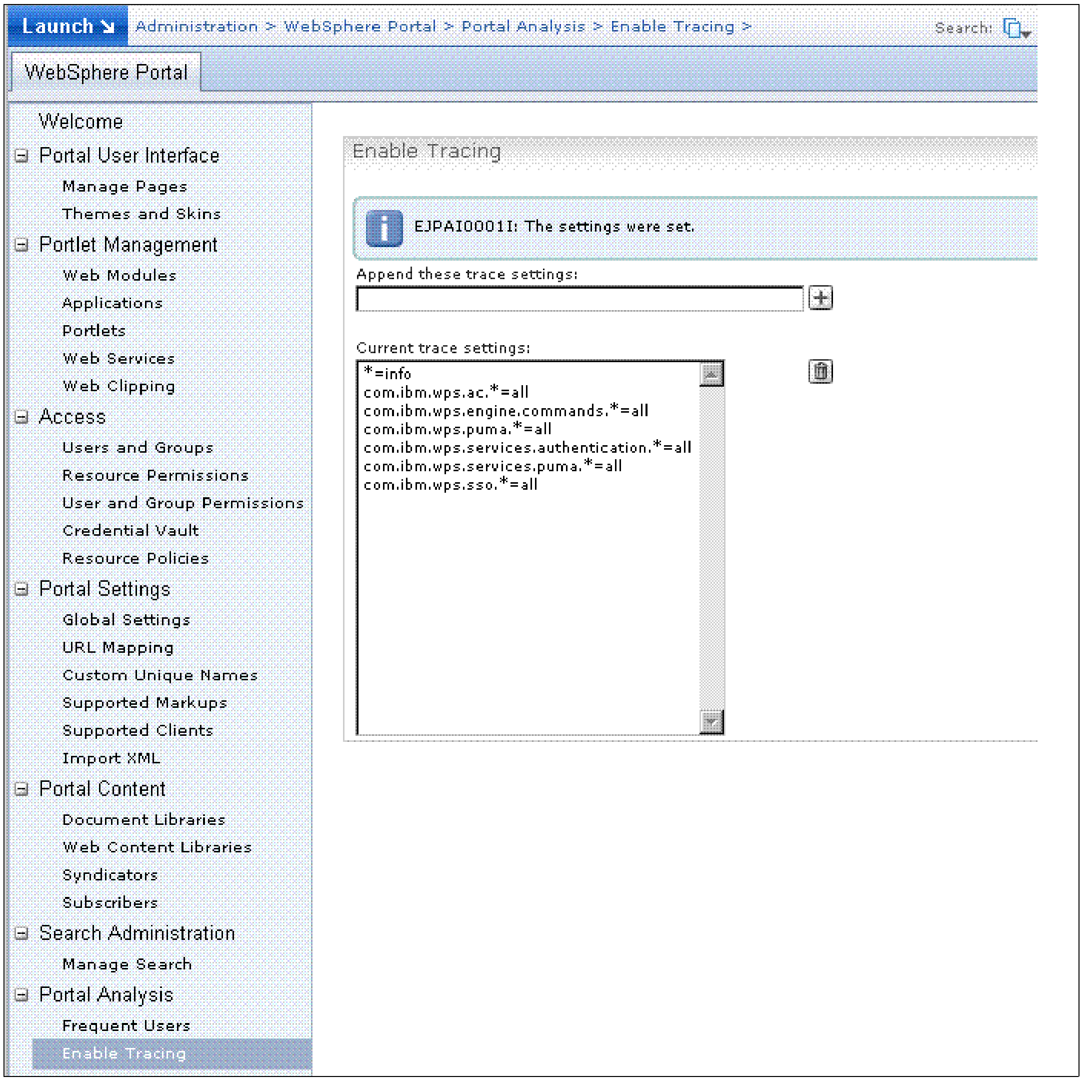
```
com.ibm.wps.engine.*=all:com.ibm.wps.puma.*=all:com.ibm.wps.services.puma.*=all:co
m.ibm.wps.sso.*=all:com.ibm.wps.services.authentication.*=all
```

In most problems related to security, we recommend using the two sets of trace strings above as a base. In the following, we refer to the combination (<wmmbase>:<authbase>) as <base>. Depending on the special scenarios you have, you may want to attach the additional strings shown in Table 4-5.

Table 4-5 Trace strings for security problems

Problem	**Trace strings**
Portal application server startup	com.ibm.ws.security.*=all without realm *<wmmbase>*:com.ibm.ws.security.*=all with realm
Single sign-on	<base>:com.ibm.ws.security.*=all
Portal login	*<base>*
WebSphere Application Server login	*<base>*:com.ibm.ws.security.*=all
Portal Access Control	*<base>*:com.ibm.wps.ac.impl.AccessControlFederator=all
Manage Users and Groups	*<base>*:com.ibm.wps.portlets.admin.*=all:com.ibm.wps.portlets.manageprincipals.*=all
TAM integration	*<base>*:com.ibm.ws.security.*=all:com.ibm.wps.ac.esm.*=all:com.ibm.wps.ac.authtable.*= all
HTTP over SSL configuration	*<base>*:com.ibm.ws.security.*=all

The traces can be enabled statically through the WebSphere Application Server's Administrative Console (under the Configuration tab), or by directly editing the file server.xml for that application server. Alternatively, the traces can also be enabled dynamically through the Runtime tab of the Admin console of the WebSphere Application Server, or the Enable Tracing portlet within WebSphere Portal administration (select **Administration** → **Portal Analysis** → **Enable Tracing**), as shown in Figure 4-6.



EJPAI0001I: The settings were set.

Append these trace settings:

Current trace settings:

```
*=info
com.ibm.wps.ac.*=all
com.ibm.wps.engine.commands.*=all
com.ibm.wps.puma.*=all
com.ibm.wps.services.authentication.*=all
com.ibm.wps.services.puma.*=all
com.ibm.wps.sso.*=all
```

Figure 4-6 Enable Tracing portlet

The static approach requires a system restart, which is not always desirable. The dynamic option is preferred under some circumstances, for example, the server cannot be brought down and restarted. The customers have to be aware of the impact on performance, however. The runtime option uses the memory buffer from the Java Virtual Machine (JVM) heap. Depending on the settings on the memory buffer, a certain amount of memory on the heap will be used for the logging. We recommend setting the traces when you want to recreate the scenario and disable them when the recreation of the scenario is completed.

When the traces are enabled statically, the trace specification should be shown at the top of the log:

```
[8/2/07 11:51:32:609 EDT] 0000000a ManagerAdmin  I   TRAS0017I: The startup trace 
state is 
*=info:com.ibm.ws.wmm.*=all:com.ibm.websphere.wmm.*=all:WSMM=all:com.ibm.ws.securi
ty.*=all:com.ibm.wps.engine.commands.*=all:com.ibm.wps.puma.*=all:com.ibm.wps.serv
ices.puma.*=all:com.ibm.wps.services.authentication.*=all:com.ibm.wps.sso.*=all.
```

When the traces are enabled dynamically, there should be a line like the following:

```
[8/21/07 9:39:14:656 EDT] 00000046 ManagerAdmin  I   TRAS0018I: The trace state 
has changed. The new trace state is 
*=info:com.ibm.ws.wmm.*=all:com.ibm.websphere.wmm.*=all:WSMM=all:com.ibm.wps.ac.*=
all.
```

Tip: The traces enabled statically can also be disabled at runtime using the admin console or the Enable Tracing portlet.

4.3.3 Tools for troubleshooting security problems

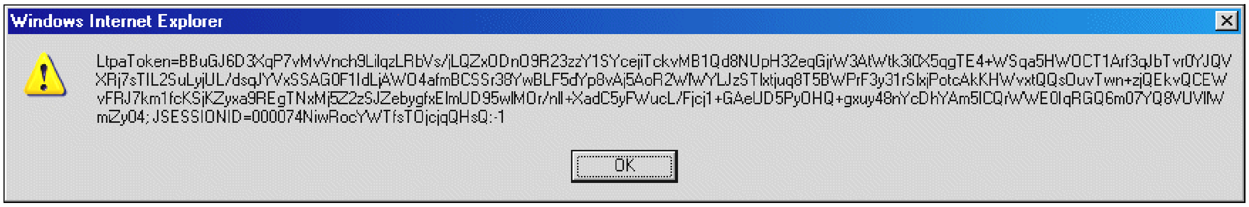
WebSphere Portal is a complex product set. To administer a site based on Portal, we assume administrators are equipped with basic LDAP knowledge:

- Understanding the basic LDAP directory structure.
- Being able to use LDAP tools, such as ldapsearch or LDAP browser, to verify user and groups, and to generate the output of a subtree, a user, or a group in LDAP Data Interchange Format (LDIF).
- Understanding the meaning and implication of the common LDAP server return codes, or at least being able to search them on the internet, such as:
  - 4 - Sizelimit exceeded
  - 10 - Referral
  - 6 - No such attribute
  - 32 - No such object
  - 49 - Invalid credentials
  - 50 - Insufficient access rights
  - 53 - Unwilling to perform.

XMLaccess is a configuration and deployment tool provided only in WebSphere Portal. Under certain circumstances, we recommend a full export using XMLaccess, especially for Portal Access Control (PAC) related issues.

To debug single sign-on or session related problems, we frequently refer to the HTTP header and cookie information. LiveHttpHeaders is a Firefox extension. It shows detailed data of what comes into the browser and what goes out. The data captured by the tool would give us a lot of debugging information about clients, cookies, protocols, URLs, and so on.

One of the often asked question is how we can see whether the browser has received the LTPA token, especially during debugging of single sign-on problems. If the browser supports JavaScript, the most straightforward way is to type `javascript:alert(document.cookie)` in the browser's location or URL field, as shown in Figure 4-7. Here you can see the LTPA token and JSESSIONID.



```
LtpaToken=BBuGJ6D3XqP7vMvVnch9LilqzLRbVs/jLQZxODnO9R23zzY1SYcejiTckvMB1Qd8NUpH32eqGjrW3AtWtk3i0X5qgTE4+WSqa5HWOCT1Arf3qJbTvr0YJQV
XRj7sTIL2SuLyjUL/dsqJYVxSSAG0F1IdLjAW04afmBCSSr38YwBLF5dYp8vAj5AoR2WMWYLJzSTlxtjuq8T5BWPrF3y31rSlxjPotcAkKHWvxtQQsOuvTwn+zjQEkvQCEW
vFRJ7km1fcKSjKZyxa9REgTNxMj5Z2zSJZebygfxEImUD95wlMOr/nll+XadC5yFWucL/Fjcj1+GAeUD5PyOHQ+gxuy48nYcDhYAm5ICQrWWE0IqRGQ6m07YQ8VUVIlW
mZy04;JSESSIONID=000074NiwRocYWTfsTOjcjqQHsQ:-1
```

Figure 4-7 LTPA token shown by "javascript:alert(document.cookie"

4.3.4 Anatomy of configuration files

Here we discuss the anatomy of the configuration files.

Configuration files for WebSphere Application Server global security

In the context of the chapter, *<portal_root>* represents the directory root where WebSphere Portal is installed. For example:

- Windows: C:\IBM\WebSphere\PortalServer
- UNIX/Linux: /opt/IBM/WebSphere/PortalServer

and *<wsas_profile_root>* is the root directory of the WebSphere Application Server profile. Depending on whether the system is standalone or in a cluster, this means two different directories. For example:

- Windows: C:\IBM\WebSphere\AppServer\profiles\wp_profile
- UNIX/Linux: /opt/IBM/WebSphere/AppServer/profiles/wp_profile

security.xml

This is the configuration file for the WebSphere Application Server global security. Whenever a security problem is encountered, this is the first file to be examined. There is only one copy of this file for a cell. Its location is at <wsas_profile_root>/config/cells/<cellname>. Do not put another copy in any of the subdirectories.

A "skeleton" of the file is shown in Example 4-3. We have omitted some of the content in the file to emphasize the information relevant to the our purposes.

Example 4-3 Sample security.xml: the first segment

```
<?xml version="1.0" encoding="UTF-8"?>
<security:Security xmi:version="2.0" ... enabled="true" cacheTimeout="600" ...
activeAuthMechanism="LTPA_1" activeUserRegistry="CustomUserRegistry_1"
defaultSSLSettings="SSLConfig_1">
...
  <authMechanisms xmi:type="security:LTPA" xmi:id="LTPA_1" OID="oid:1.3.18.0.2.30.2"
authContextImplClass="com.ibm.ISecurityLocalObjectTokenBaseImpl.WSSecurityContextLTPAImpl"
authConfig="system.LTPA" simpleAuthConfig="system.LTPA" authValidationConfig="system.LTPA"
timeout="480" password="{xor}KzYyOms5KjE=">
    <trustAssociation xmi:id="TrustAssociation_1" enabled="false">
      <interceptors xmi:id="TAInterceptor_1"
interceptorClassName="com.ibm.ws.security.web.WebSealTrustAssociationInterceptor"/>
```

```
      <interceptors xmi:id="TAInterceptor_2"
interceptorClassName="com.ibm.ws.security.web.TAMTrustAssociationInterceptorPlus"/>
    </trustAssociation>
    <singleSignon xmi:id="SingleSignon_1" requiresSSL="false" domainName="acme.com"
enabled="true"/>
    <private xmi:id="Key_1174328477781"
byteArray="d7zRPA3tyjvF5+XsCyCDPHR4OaV4cIIrPOY1xrhjpjwyEUJBrXSPrD3psTZL9r4e22JcFh1BYjMO8FrF2TaCq
sxbBc6j442UklMqgiGxWt+OA6MtEzdZT2cR/1HR2efFved19BNFp4KgNYEvXOMXhbUtpIr4arXXgiJPCoTdds6FNojLeiUcA
DdVVsUOzZAu24c6yfE4mkyZEzgUPmpw4sHIuSA+g1zvx1cGP8VqSZuKuOLfBLQlKITqEIHyvzovEhi6Jdf7serNG/bKUINwa
zkCLqcD3vTTKynOQDD7sr99AjGJO9ZKJCjUTChs1K4PKMiw9AAiPlPkyelaJZIeLG/Ml5NH+Dk2zBu9h9eqkmw="/>
    <public xmi:id="Key_1174328477782"
byteArray="ALKQhPAflWyZ6vtTrrwirLBDOKHelxK3T7V/lH5WwOisL35ogUw3cThpSJ9eCH4BZcOwOazNjPsXGNJOyf1aV
BpmtqS2fTcif9I6Olh7FZPrqr3leFfli7ioOCtEmIK6iq9+pOtTyw2yB+IWMmKeABK5OyUr8xLylqx9TyiO1E3XAQAB"/>
    <shared xmi:id="Key_1174328477783"
byteArray="eaV+DxdLHfpiCqWwCbEOedPLNvvFrUAZAYLN1eDakpO="/>
  </authMechanisms>
```

The very first enabled in the file marks whether the global security is enabled. In some cases, you can manually set it to false to temporarily disable security so that you can start the servers.

activeUserRegistry specifies that the actual user registry is configured. It can tell whether the security was enabled to support WMM realm (with WMMUR) or not (LDAP only). The Custom User Registry entries that appear later in the example give the configuration details of WMMUR. Notice that the file locations in a cluster are different. They must point to those under *<wsas_profile_root>*/config/wmm.

The trustAssociation stanza defines all the definitions of all the Trust Association Interceptors. The TAI would be loaded only when enabled is set to true.

The single sign-on (SSO) domain is required in most cases. As we have explained in 4.1.4, “Single sign-on (SSO)” on page 90, you can leave the domain blank, but this would only work with a single system case.

The LTPA key is given as a private, a public, and a shared key. Whenever you generate a new LTPA key, the three values in this file change. When multiple WebSphere application servers participate in a single sign-on domain, they should share the same LTPA key and the three values shown in security.xml should be exactly the same.

Example 4-4 Sample security.xml: the second segment

```
<userRegistries xmi:type="security:LocalOSUserRegistry" xmi:id="LocalOSUserRegistry" serverId=""
serverPassword="{xor}" realm=""/>
  <userRegistries xmi:type="security:CustomUserRegistry" xmi:id="CustomUserRegistry_1"
serverId="uid=wasadmin,ou=people,ou=dept,o=acme.com" serverPassword="{xor}KD4sPjsyNjE="
ignoreCase="true" realm="corpldap.acme.com:389"
customRegistryClassName="com.ibm.websphere.wmm.registry.WMMUserRegistry">
    <properties xmi:id="Property_1174328488359" name="WMMUR_LOGGING" value="true"
required="false"/>
    <properties xmi:id="Property_1174328488906" name="WMMUR_CONFIG"
value="C:/IBM/WEBSPH~1/PORTAL~1/wmm/wmmur.xml" required="true"/>
    <properties xmi:id="Property_1174328489234" name="WASUSER_REGISTRY_TYPE"
value="wmmFileRegistry" required="false"/>
    <properties xmi:id="Property_1174328489672" name="wmmUserSecurityNameAttr" value="uid"
required="true"/>
```

```
    <properties xmi:id="Property_1174328490172" name="wasAdminFileLoc"
value="C:/IBM/WEBSPH~1/PORTAL~1/wmm/wmmWASAdmin.xml" required="true"/>
    <properties xmi:id="Property_1186336290766" name="userRegistryRealm"
value="corpldap.acme.com:389" required="false"/>
  </userRegistries>
  <userRegistries xmi:type="security:LDAPUserRegistry" xmi:id="LDAPUserRegistry_1"
serverId="uid=wasadmin,ou=people,ou=dept,o=acme.com" serverPassword="{xor}HB8rEW8aHyO\="
realm="corpldap.acme.com:389" ignoreCase="true" type="IBM_DIRECTORY_SERVER" sslEnabled="false"
sslConfig="wp6vm_n/DefaultSSLSettings" baseDN="uid=wasadmin,ou=people,ou=dept,o=acme.com"
bindDN="" bindPassword="{xor}HB8rEW8aHyO\=" searchTimeout="120" reuseConnection="true">
    <searchFilter xmi:id="LDAPSearchFilter_1"
userFilter="(&amp;(uid=%v)(objectclass=inetOrgPerson))"
groupFilter="(&amp;(cn=%v)(objectclass=groupOfUniqueNames))" userIdMap="*:uid" groupIdMap="*:cn"
groupMemberIdMap="ibm-allGroups:uniqueMember" certificateMapMode="EXACT_DN"
certificateFilter=""/>
    <hosts xmi:id="EndPoint_1" host="corpldap.acme.com" port="389"/>
  </userRegistries>
```

The active user registry is highlighted and its ID is specified by the activeRegistry parameter at the beginning. People sometimes are confused about which registry is configured. In this example, we can see that WMMUR (CustomUserRegistry) is active. We can also find some configuration information in the LDAPUserRegistry section. This tells us that the administrator might have configured the LDAP without realm support before and the LDAP related configuration remains in the file. This may not be necessarily bad. We should simply be aware which registry is *active*.

Note: Running the configuration task “disable-security” does not erase the configuration settings in the global security configuration of WebSphere Application Server. It simply sets enabled to false.

The user registry realm and customer property userRegistryRealm defined in the WMMUR segment should point to the same LDAP server and port. These configurations are required to be manually added for working with other application servers, such as Domino, for single sign-on (SSO).

Tip: Do not confuse the user registry realm with the WMM realm defined in wmmur.xml. The realm defined here is only to identify the LDAP realm for single sign-on. It has nothing to do with the separation of user populations used in WebSphere Portal’s virtual portals.

Tip: The location of the WMMUR configuration files in a cluster is different. It is based on the WebSphere variable ${WMM_CONFIG_PATH} created during cluster creation.

Example 4-5 shows the third segment of the sample security.xml file.

Example 4-5   Sample security.xml: the third segment

```
...
  <applicationLoginConfig xmi:id="JAASConfiguration_1">
    <entries xmi:id="JAASConfigurationEntry_1" alias="ClientContainer">
      <loginModules xmi:id="JAASLoginModule_1"
moduleClassName="com.ibm.ws.security.common.auth.module.proxy.WSLoginModuleProxy"
authenticationStrategy="REQUIRED">
```

```
        <options xmi:id="Property_1" name="delegate" 
value="com.ibm.ws.security.common.auth.module.WSClientLoginModuleImpl"/>
      </loginModules>
    </entries>
    <entries xmi:id="JAASConfigurationEntry_2" alias="WSLogin">
      <loginModules xmi:id="JAASLoginModule_2" 
moduleClassName="com.ibm.ws.security.common.auth.module.proxy.WSLoginModuleProxy" 
authenticationStrategy="REQUIRED">
        <options xmi:id="Property_2" name="delegate" 
value="com.ibm.ws.security.common.auth.module.WSLoginModuleImpl"/>
        <options xmi:id="Property_3" name="use_realm_callback" value="false"/>
        <options xmi:id="Property_4" name="use_appcontext_callback" value="false"/>
      </loginModules>
    </entries>
    <entries xmi:id="JAASConfigurationEntry_3" alias="DefaultPrincipalMapping">
      <loginModules xmi:id="JAASLoginModule_3" 
moduleClassName="com.ibm.ws.security.common.auth.module.proxy.WSLoginModuleProxy" 
authenticationStrategy="REQUIRED">
        <options xmi:id="Property_5" name="delegate" 
value="com.ibm.ws.security.auth.j2c.WSPrincipalMappingLoginModule"/>
      </loginModules>
    </entries>
    <entries xmi:id="JAASConfigurationEntry_1174328490828" alias="Portal_WSRP_Login"/>
    <entries xmi:id="JAASConfigurationEntry_1174328491438" alias="Portal_Login"/>
    <entries xmi:id="JAASConfigurationEntry_1174328492250" alias="Portal_LTPA">
      <loginModules xmi:id="JAASLoginModule_1174328492594" 
moduleClassName="com.ibm.ws.security.common.auth.module.proxy.WSLoginModuleProxy" 
authenticationStrategy="REQUIRED">
        <options xmi:id="Property_1174328492656" name="delegate" 
value="com.ibm.ws.security.server.lm.ltpaLoginModule" required="true"/>
      </loginModules>
      <loginModules xmi:id="JAASLoginModule_1174328492875" 
moduleClassName="com.ibm.ws.security.common.auth.module.proxy.WSLoginModuleProxy" 
authenticationStrategy="REQUIRED">
        <options xmi:id="Property_1174328492891" name="delegate" 
value="com.ibm.ws.security.server.lm.wsMapDefaultInboundLoginModule" required="true"/>
        <options xmi:id="Property_1174328492922" name="cookie" value="true" required="true"/>
      </loginModules>
    </entries>
  </applicationLoginConfig>

...
  <repertoire xmi:id="SSLConfig_1" alias="wp6vm_n/DefaultSSLSettings">
    <setting xmi:id="SecureSocketLayer_1" 
keyFileName="${USER_INSTALL_ROOT}/etc/DummyServerKeyFile.jks" keyFilePassword="{xor}CDo9Hgw=" 
keyFileFormat="JKS" trustFileName="${USER_INSTALL_ROOT}/etc/DummyServerTrustFile.jks" 
trustFilePassword="{xor}CDo9Hgw=" trustFileFormat="JKS" clientAuthentication="false" 
securityLevel="HIGH" enableCryptoHardwareSupport="false">
      <cryptoHardware xmi:id="CryptoHardwareToken_1" tokenType="" libraryFile="" 
password="{xor}"/>
      <properties xmi:id="Property_6" name="com.ibm.ssl.protocol" value="SSL"/>
      <properties xmi:id="Property_7" name="com.ibm.ssl.contextProvider" value="IBMJSSE2"/>
    </setting>
  </repertoire>
...
```

```
  <authDataEntries xmi:id="JAASAuthData_1174051597218" alias="wp6vm_c/samples" userId="samples"
password="{xor}LG4+Mi8zOiw=" description="JAAS Alias for WebSphere Samples"/>
  <authDataEntries xmi:id="JAASAuthData_1174052349281" alias="wpdbDSJAASAuth" userId="db2admin"
password="{xor}DTovMz48Ogg2KzcGMCotGz0eOzI2MQ8oOw==" description="JAAS Alias for DataSource
wpdbDS"/>
  <authDataEntries xmi:id="JAASAuthData_1174052419453" alias="designerDSJAASAuth"
userId="db2admin" password="{xor}DTovMz48Ogg2KzcGMCotGz0eOzI2MQ8oOw==" description="JAAS Alias
for DataSource designerDS"/>
  <authDataEntries xmi:id="JAASAuthData_1174052423750" alias="syncDSJAASAuth" userId="db2admin"
password="{xor}DTovMz48Ogg2KzcGMCotGz0eOzI2MQ8oOw==" description="JAAS Alias for DataSource
syncDS"/>
...
              </security:Security
```

The applicationLoginConfig entries define the JAAS login modules, and the Portal_LTPA mentioned in 4.1.5, “WebSphere Portal login process” on page 91 is one of the entries. When you extend the Portal login process, the Portal_Login module will contain the Java class name.

The repertoire entries contain SSL repertoires defined for Secure Socket Layer communication. If you customize the key and trust files, the file names and paths must be replaced here.

The authDataEntries specify the J2A authentication aliases for accessing the datasources defined in JDBC providers at runtime.

admin-authz.xml

This file is in the same directory as security.xml. It contains the users and groups for the administrative console administration. Example 4-6 shows the content of a sample.

Example 4-6 Sample admin-authz.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<rolebasedauthz:AuthorizationTableExt xmi:version="2.0" xmlns:xmi="http://www.omg.org/XMI"
xmlns:rolebasedauthz="http://www.ibm.com/websphere/appserver/schemas/5.0/rolebasedauthz.xmi"
xmi:id="AuthorizationTableExt_1" context="domain">
  <authorizations xmi:id="RoleAssignmentExt_1" role="SecurityRoleExt_1">
    <users xmi:id="UserExt_1109285497219" name="cn=wpsbind,ou=people,ou=dept,o=acme.com"/>
    <users xmi:id="UserExt_1142530744703" name="cn=wpsadmin,ou=people,ou=dept,o=acme.com"/>
<specialSubjects xmi:type="rolebasedauthz:ServerExt" xmi:id="ServerExt_1"/>
  </authorizations>
  <authorizations xmi:id="RoleAssignmentExt_2" role="SecurityRoleExt_2"/>
  <authorizations xmi:id="RoleAssignmentExt_3" role="SecurityRoleExt_3"/>
  <authorizations xmi:id="RoleAssignmentExt_4" role="SecurityRoleExt_4">
    <users xmi:id="UserExt_1157057598297" name="cn=asdf,ou=people,ou=dept,o=acme.com"/>
  </authorizations>
  <roles xmi:id="SecurityRoleExt_1" roleName="administrator"/>
  <roles xmi:id="SecurityRoleExt_2" roleName="operator"/>
  <roles xmi:id="SecurityRoleExt_3" roleName="configurator"/>
  <roles xmi:id="SecurityRoleExt_4" roleName="monitor"/>
</rolebasedauthz:AuthorizationTableExt>
```

You can see that two users, wpsbind and wpsadmin, were assigned the Administrator role and the Monitor role.

WebSphere Member Manager (WMM) configuration files

The main configuration files for the WebSphere Member Manager (WMM) are inside the directory *<portal_root>*/wmm, which is outside of the scope of the WebSphere Application Server. In a clustered environment, in order for the Deployment Manager (Dmgr) to be able to synchronize the files with the nodes in the cell, these WMM files are copied into *<wsas_profile_root>*/config/cells/wmm. Thus, when making changes to the WMM configuration files in a clustered environment, you should not simply modify the files on the node or Dmgr directly; instead, the recommended and supported process is shown below:

1. From the primary node, change the directory to <portal_root>/config, and run the following configuration task to check out the WMM files from Dmgr:
   - UNIX/Linux:./WPSconfig.sh check-out-wmm-cfg-files-from-dmgr
   - Windows: WPSconfig.bat check-out-wmm-cfg-files-from-dmgr
2. Change the directory to <portal_root>/wmm on the primary node, and modify the WMM files in the directory.
3. Change the directory to <portal_root>/config, and run the following command:
   - UNIX/Linux:./WPSconfig.sh check-in-wmm-cfg-files-to-dmgr
   - Windows: WPSconfig.bat check-in-wmm-cfg-files-to-dmgr
4. Run a full manual synchronization from the Dmgr to push the changes to all nodes.
5. Restart the cluster to make the change effective.

Note: The WMM files are read once only during server startup. A restart is required for any changes made to them to be effective.

wmm.xml

This is the most important file for WMM configuration. Any typo in the file can prevent the WMM EJB from starting up or functioning correctly. As we already stated in Example 1, when making manual changes to this file in a clustered environment, only modify the one under <portal_root>/wmm on the primary node, and use the “check-out” and “check-in” procedure to keep the file synchronized to the Deployment Manager and other nodes. This file specifies configuration settings for WMM, such as the supported member types (Person, Group, OrganizationalUnit, and Organization), the LDAP server host name and the bind user DN and password, WMM connection pool data, different repositories, and so on. A scaled-down sample of wmm.xml is shown in Example 4-10 on page 118.

Example 4-7 A sample wmm.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<wmm name="member manager"
   defaultRealmName="portal"
   horizontalPartitioning="false"
   lookAside="true"
   configurationFile="wmmAttributes.xml"
   maximumSearchResults="200"
   searchTimeOut="120000"
   userSecurityNameAttribute="uid"
   passwordAttribute="userPassword">

   <supportedMemberTypes>
      <supportedMemberType name="Person"
         rdnAttrTypes="uid"
         defaultParentMember="ou=people,ou=dept,o=acme.com"
      defaultProfileRepository="LDAP1"/>
   <supportedMemberType name="Group"
      rdnAttrTypes="cn"
      defaultParentMember="ou=groups,ou=dept,o=acme.com"
      defaultProfileRepository="LDAP1"/>
</supportedMemberTypes>

<repositories>
   <lookAsideRepository name="wmmDBLookAside"
      UUID="LA"
      adapterClassName="com.ibm.ws.wmm.lookaside.db.LookAsideAdapter"
      supportDynamicAttributes="true"
      dataSourceName="jdbc/wpdbDS"
      databaseType="db2"
      dataAccessManagerClassName="com.ibm.ws.wmm.db.dao.db2.WMMCloudscapeDao"/>

   <ldapRepository name="wmmLDAP"
      UUID="LDAP1"
      adapterClassName="com.ibm.ws.wmm.ldap.ibmdir.IBMDirectoryAdapterImpl"
      configurationFile="wmmLDAPServerAttributes.xml"
      profileRepositoryForGroups="LDAP1"
      adminId="uid=bindid,ou=people,ou=dept,o=acme.com"
      adminPassword="afacWLqg1trlbNupQsppiw=="
      ldapHost="corpldap.acme.com"
      ldapPort="389"
      ldapType="0"
      sslEnabled="false"
      sslTrustStore="C:\WebSphere\AppServer\etc\DummyServerTrustFile.jks"
      dirContextsMaxSize="20"
      dirContextsMinSize="5"
      dirContextTimeToLive="-1"
      cacheGroups="false"
      groupsCacheTimeOut="600"
      cacheAttributes="true"
      attributesCacheSize="2000"
      attributesCacheTimeOut="600"
      cacheNames="true"
      namesCacheSize="2000"
      namesCacheTimeOut="600">

      <nodeMaps>
         <nodeMap node="ou=people,ou=dept,o=acme.com"
         pluginNode="ou=people,ou=dept,o=acme.com"/>

         <nodeMap node="ou=groups,ou=dept,o=acme.com"
         pluginNode="ou=groups,ou=dept,o=acme.com"/>
      </nodeMaps>

      <supportedLdapEntryTypes>
         <supportedLdapEntryType name="Person"
         rdnAttrTypes="uid"
         objectClassesForRead="inetOrgPerson"
         objectClassesForWrite="inetOrgPerson"
         searchBases="ou=people,ou=dept,o=acme.com"/>
```

```
            <supportedLdapEntryType name="Group"
            rdnAttrTypes="cn"
            objectClassesForRead="groupOfUniqueNames"
            objectClassesForWrite="groupOfUniqueNames"
            searchBases="ou=groups,ou=dept,o=acme.com"/>
         </supportedLdapEntryTypes>
      </ldapRepository>
   </repositories>
</wmm>
```

Within the WMM configuration, the default realm name is set to portal. If you prefer a different name, you can choose one and set it to “WmmDefaultRealm” in wpconfig.properties, and then run the security configuration task, or you can change it after the security is enabled, by modifying defaultRealmName in wmm.xml, and the name of “default” realm in wmmur.xml.

maximumSearchResults is the parameter associated with the search requests WMM sent to the LDAP server, if there is no size limit set up on the LDAP server. When a sizelimitExceededException is found in runtime or the trace log files, you can try to increase the value. The recommendation is not to set it up too high. If the returned search result becomes too high, the impact on performance on the LDAP would be high. In such cases, we recommend either a better search filter or a narrow search base.

userSecurityAttributeName should always be set to the login attribute. This attribute defaults to the Relative Distinguished Name (RDN™) in most cases, but it is not necessary. When WMMUR is configured, this should be the same as the customer property wmmUserSecurityNameAttr (refer to Example 4-3 on page 110).

ldapHost and ldapPort should be set to the same host name and port configured in the WebSphere Application Server security configuration.

Since the member types OrganizationalUnit and Organization are not used in WebSphere Portal, the support for them can be dropped. If you opt out of doing that, the corresponding entries should be removed from wmmAttributes.xml and wmmLDAPServerAttributes.xml or wmmDBAttributes.xml as well.

wmmAttributes.xml

This is the dictionary of all attributes used for WMM members. It is used as a reference for all attributes used within the portal environment. The only exception is extId, which is an internal attribute, and it is not supposed to be modified by applications. An example attribute entry is shown in Example 4-8.

Example 4-8 An example of attribute definition in wmmAttributes.xml

```
<attribute wmmAttributeName="uid"
   applicableMemberTypes="Person"
   requiredMemberTypes="Person"
   dataType="String"
   valueLength="254"
   multiValued="false"/>
```

Notice that requiredMemberTypes specifies that this attribute is mandatory for member type Person. If an attribute is multi-valued, that is, multiValued set to true, the values are separated by semicolons.

wmmLDAPServerAttributes.xml

This file maps the WMM attribute reference names to the actual attribute names in LDAP server. The WMM attribute name is like a logical name and used in the calls to WMM. They can be different from the ones used in the LDAP server. The typical entry is like the one shown in Example 4-9.

Example 4-9 WMM attribute map in wmmLDAPServerAttributes.xml

```
<attributeMap wmmAttributeName="uid"
   applicableMemberTypes="Person"
   pluginAttributeName="samAccountName"
   dataType="String"
   valueLength=”32”
   multiValued="false"/>
```

In this example, attribute uid is mapped to samAccountName in Microsoft Active Directory. This attribute is applicable for member type “Person” only. pluginAttributeName specifies the real attribute name in the LDAP server, in this case, Microsoft Active Directory.

wmmur.xml

With realm support, this file defines all realms used in Portal. The content of this file is not populated by any of the configuration tasks (enable-security-wmmur-ldap, enable-security-wmmur-db, or enable-security-wmmur-custom). It must be set up manually by the Portal administrator after the security is configured. An example of this file is shown in Example 4-10.

Example 4-10 A sample wmmur.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<wmmur>
   realms>
      <realm id="portal" delimiter="@" default="true">
         node wmmnode="ou=dept,o=acme.com"/>
         node wmmnode="ou=people,ou=dept,o=acme.com" defaultParent="Person"/>
         node wmmnode="ou=groups,ou=dept,o=acme.com" defaultParent="Group"/>
      /realm>
      realm id="internet" delimiter="@" default="false">
         node wmmnode=”uid=wpsadmin,ou=people,ou=dept,o=acme.com”/>
         node wmmnode="dc=loc1,dc=abc,dc=com"/>
         <node wmmnode="cn=users,dc=internet,dc=abc,dc=com"
            defaultParent="Person"/>
         node wmmnode="cn=groups,dc=internet,dc=abc,dc=com"
            defaultParent="Group"/>
      /realm>
      <realm id="intranet" delimiter="@" default="false">
         node wmmnode=”uid=wpsadmin,ou=people,ou=dept,o=acme.com”/>
         node wmmnode="dc=loc2,dc=abc,dc=com"/>
         node wmmnode="cn=users,dc=intranet,dc=abc,dc=com"
            defaultParent="Person"/>
         node wmmnode="cn=groups,dc=intranet,dc=abc,dccom" defaultParent="Group"/>
      /realm>
   /realms>
</wmmur>
```

Only one default realm is allowed. Make sure you have only one set to true.

Note: The base (root) portal and portal admin user (“wpsadmin”) must be configured in the default realm.

Note: Each of the wmmnode entries in wmmur.xml must correspond to a nodeMap entry in wmm.xml. This is how WMM ties the realm definition of user populations in this file with general WMM configurations together.

Make sure the location of this pointer in the cluster configuration is correct.

In order for the admin user to be able to access the virtual portals configured with the realms defined, we recommend adding the admin user “wpsadmin” to every realm, as shown in the example.

wmmWASAdmin.xml

When the security is enabled with realm support, the normal way of controlling the access to the Administrative Console of WebSphere Application Server is through file wmmWASAdmin.xml. After a console user is added through the administrative console, you have to manually add an entry for that user into this file. This file only contains users, not groups. This means that there is no equivalent support for console groups using this file. The format of this file is shown in Example 4-11.

Example 4-11 wmmWASAdmin.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<wmmWASAdmins>
   <admin logonId="uid=wasadmin,ou=people,ou=dept,o=acme.com"
      logonPassword="anvu7zPZ7jbrZLa4h89Tfg=="
      uniqueUserId="uid=wasadmin,ou=people,ou=dept,o=acme.com"/>
   <admin logonId="wasadmin"
      logonPassword="anvu7zPZ7jbrZLa4h89Tfg=="
      uniqueUserId="uid=wasadmin,ou=people,ou=dept,o=acme.com"/>
</wmmWASAdmins>
```

Although one can use a plain-text password in this file, we strongly recommend encrypting the password using the WMM utility called **`wmm_encrypt.bat/.sh`**. An alternative to this approach of manually modifying the file wmmWASAdmin.xml using an editor is using the utility **`updateWmmWASAdminRegistry.bat/.sh`**. IBM TechNote 1246919 had documented the usage of this utility.

The second entry for the same user ID wasadmin allows using the short user ID when starting the application servers and logging in on the Administrative console.

4.3.5 Reading portal runtime logs

In WebSphere Portal Version 6, the runtime activities are recorded in the application server’s Java Virtual Machine (JVM) logs, that is, SystemOut.log and SystemErr.log. For any problems reported during runtime, these logs should be reviewed first. During WebSphere Portal configuration activities, when either using the configuration command-line utility (WPSconfig.sh/.bat) or the graphic configuration wizard, all the information is captured to ConfigTrace.log. When something unexpected happens, these are the files to be reviewed for clues. When diagnostics traces are enabled, trace.log should be reviewed with the reference to the JVM runtime log files to show the correlations of the events by matching their timestamps.

Application server startup

The first thing to look for is whether there are any exceptions. Not all exceptions are critical to the portal server. Some of them are due to the nature of multithreads of the application server, and can be ignored. As long as the key services of WebSphere Application Server and WebSphere Portal are started, the system should normally be fine. To find the set of the WebSphere Application Server and WebSphere Portal key services, refer to the Information Centers of these two products.

The security configuration would be loaded early as shown in the log, and the following two lines show that it is successfully loaded:

```
[8/2/07 11:52:24:734 EDT] 0000000a distSecurityC I   SECJ0243I: Security service
started successfully
[8/2/07 11:52:24:750 EDT] 0000000a distSecurityC I   SECJ0210I: Security enabled
true
```

WMM startup

wmmApp is the WMM Enterprise Java Been (EJB) installed on the Application Server. Its successful startup is the first step for Portal security to function. If it fails to start, Portal security will not work. The typical sequence of the startup in SystemOut.log should look like the following:

```
[4/30/07 16:15:54:429 PDT] 0000000a ApplicationMg A   WSVR0200I: Starting
application: wmmApp
[4/30/07 16:15:55:728 PDT] 0000000a EJBContainerI I   WSVR0207I: Preparing to
start EJB jar: wmm.ejb.jar
[4/30/07 16:15:55:763 PDT] 0000000a EJBContainerI I   WSVR0037I: Starting EJB jar:
wmm.ejb.jar
[4/30/07 16:15:55:931 PDT] 0000000a EJBContainerI I   WSVR0207I: Preparing to
start EJB jar: wmm.internal.ejb.jar
[4/30/07 16:15:55:953 PDT] 0000000a EJBContainerI I   WSVR0037I: Starting EJB jar:
wmm.internal.ejb.jar
[4/30/07 16:15:55:995 PDT] 0000000a ApplicationMg A   WSVR0221I: Application
started: wmmApp
```

Sometimes even though wmmApp is shown as started, WMM could still fail as follows:

```
[8/14/07 15:11:29:188 EDT] 0000000a WSMM Message  E
com.ibm.ws.wmm.MemberRepositoryManager init Initialization failed due to invalid
property "supportedMemberTypes".
[8/14/07 15:11:29:203 EDT] 0000000a WSMM Message  E
com.ibm.ws.wmm.objectimpl.MemberServiceBeanBase ejbCreate()
com.ibm.websphere.wmm.exception.InitializationException: Initialization failed due
to invalid property "supportedMemberTypes".
[8/14/07 15:11:29:234 EDT] 0000000a ExceptionUtil E   CNTR0019E: EJB threw an
unexpected (non-declared) exception during invocation of method
"getConfigurationData".
```

In this case, a typo in wmm.xml was the cause of the error. The failure of wmmApp would definitely cause PumaService in WebSphere Portal to fail, resulting in the failure of the portal server itself.

Adminconsole application startup

Unlike Version 5.1, the adminconsole application now is running on the application server, WebSphere_Portal, using the default port 10027. The following lines identify its successful startup:

```
[8/2/07 11:53:00:672 EDT] 0000000a ApplicationMg A   WSVR0200I: Starting
application: adminconsole
[8/2/07 11:53:08:781 EDT] 0000000a WebGroup      A   SRVE0169I: Loading Web
Module: adminconsole.
[8/2/07 11:53:11:125 EDT] 0000000a ServletWrappe A   SRVE0242I: [adminconsole]
[/ibm/console] [EventInitializer]: Initialization successful.
```

Once you see these lines, you can access the adminconsole, even though the portal server is not yet started.

Portal startup

The following line in the log signals the startup of the portal servlet:

```
[8/2/07 11:53:21:625 EDT] 0000000a WebGroup      A   SRVE0169I: Loading Web
Module: WebSphere Portal Server.
```

There is a misconception that the following line in SystemOut.log signals the Portal startup:

```
[8/2/07 11:58:29:609 EDT] 0000000a WsServerImpl  A   WSVR0001I: Server
WebSphere_Portal open for e-business
```

As a matter of fact, it only tells us that the application server is started. It does not mean that the Portal servlet has actually started. The applications and services will be initialized after the application server is successfully started. Failure of one or more portal services could result in the failure of the portal servlet. Usually, failure of one or more individual portlet applications would not affect the entire portal server, but some may affect the usage of the server, such as the Login portlet. Failure of the Login portlet would prevent the normal login process. As described in 4.1.5, “WebSphere Portal login process” on page 91, there are still other ways to access the system, which in some cases are the ways to correct the problem of the Login portlet.

The failure of PumaService shown below was caused by the WMM initialization failure mentioned in “WMM startup” on page 120:

```
[8/14/07 15:11:29:672 EDT] 0000000a Servlet       E com.ibm.wps.engine.Servlet
init EJPFD0016E: Initialization of service failed.
      com.ibm.wps.util.DataBackendException: EJPSG0015E: Data Backend Problem
java.rmi.ServerException: RemoteException occurred in server thread; nested
exception is:
   com.ibm.ejs.container.CreateFailureException: ; nested exception is:
   java.lang.reflect.InvocationTargetException
   at
com.ibm.wps.services.puma.RealmAwareURManager.initRealms(RealmAwareURManager.java:
152)
   at
com.ibm.wps.services.puma.RealmAwareURManager.<init>(RealmAwareURManager.java:91)
   at com.ibm.wps.services.puma.PumaServiceImpl.init(PumaServiceImpl.java:184)
(more stacktrace)
[8/14/07 15:11:29:750 EDT] 0000000a WebExtensionP W   Servlet portal is currently
unavailable: Initialization of one or more services failed.
[8/14/07 15:11:29:766 EDT] 0000000a WebExtensionP E   SRVE0026E: [Servlet
Error]-[javax.servlet.UnavailableException: Initialization of one or more services
failed.
```

Since PumaService is the base for WebSphere Portal security, its failure would cause the portal server to fail.

ConfigTrace.log

This file contains important messages for all configuration tasks. It should never be discarded.

The Portal configuration command-line tool **`WPSconfig.bat/.sh`** is an extension of the Apache Ant build tool. It calls Java classes, Database SQL scripts, Jacl scripts through **`wsadmin`** (a WebSphere command-line utility), or an XMLaccess command-line utility to configure various parts of WebSphere Portal and WebSphere Application Server. It signals the success or failure of a configuration task or actions by issuing either a BUILD SUCCESSFUL or BUILD FAILED message at the end.

Typical error messages in runtime logs

All portal error message codes are in the format of “EJPxxnnnnE”. Normally, when you see a portal error, there may be some errors from the WebSphere Application Server side.

4.3.6 Typical security configuration problems

In this section, we provided several typical problem scenarios. This is not intended to be a complete set of all security problems. We only want to provide several cases that are often encountered and provide suggestions on how to try to resolve them.

Cannot start servers after security was enabled

The two most likely causes of startup failure are due to database connection or a configuration in security. For this section, we only concentrate on the latter. In most cases, the failure is due to the failed authentication of the WebSphere Application Server administration user.

Using LDAP tools, like an LDAP browser or ldapsearch, try to verify that the LDAP bind user is able to access the LDAP, and the WebSphere Application Server admin user’s DN and password is correct. Then check the baseDN (LDAP suffix) and user filter definition in WebSphere Application Server global security are set correctly. Since the server is not started, you may have to edit the security settings directly in security.xml. With WebSphere Application Server, security trace enabled,

When the server could not be started, it is likely that the WebSphere Admin user had failed at authentication. If the password might have been changed, or the company security policy had required the password to be changed, this would cause the authentication failure of the WebSphere administrative user ID, as shown in the following case:

```
[8/12/07 15:32:12:672 EDT] 00000017 WSMM Message  E
com.ibm.ws.wmm.ldap.LdapConnectionImpl void initialize(Map envProperties)
Initialization failed. Root cause is:
"javax.naming.AuthenticationException: [LDAP: error code 49 - 80090308: LdapErr:
DSID-0C090334, comment: AcceptSecurityContext error, data 52e, vece]".
[8/12/07 15:32:12:812 EDT] 00000017 WSMM Message  E
com.ibm.ws.wmm.objectimpl.MemberServiceBeanBase ejbCreate()
java.lang.NullPointerException
[8/12/07 15:32:12:875 EDT] 00000017 ExceptionUtil E   CNTR0019E: EJB threw an
unexpected (non-declared) exception during invocation of method
"getConfigurationData". Exception data:
com.ibm.ejs.container.CreateFailureException: ; nested exception is:
java.lang.reflect.InvocationTargetException
```

If the bind user has the password problem with the LDAP server, the access to the LDAP server might be prohibited and the authentication would also fail. In this case, you may see `LDAP error code: Insufficient Access Rights` in the log.

Login failure

Imagine that the system was working fine. For some reason, the login suddenly fails. If there is a known change in the configuration, the first thing to try is to revert the change and test whether that resolves the problem. If it does, then the configuration change must be reviewed. Sometimes some unknown changes not in your control disrupts the system, and a systematic problem determination process is needed to try to isolate the problem, eliminate the potential factors, and narrow down the possible paths to finally find the solution.

Step 1: Understand the problem

If this is a new system and the failed user is the administrator user, try to log in on both the WebSphere Application Server console and the WebSphere Portal using the respective user IDs. If only the portal login fails, then there may be a configuration issue in the single sign-on configuration. If both fail, then the configuration settings should be closely reviewed.

If the user short name fails, but the full user DN can log in, then there may be a configuration problem with the user filter or search base.

If only a few users have problems, and others are OK, find the differences between these few users and others. When this problem is only intermittent, compare the success and failure cases, such as the clients used, access URLs, time of the day, and so on.

If there are recent configuration changes on the portal server, the LDAP server, the database, or network, try to revert the change back and see whether it resolves the issue. For example:

- If a custom login portlet is used, try the Login portlet bundled with WebSphere Portal.
- If a custom theme is used, try the default WebSphere them.

Try to find whether a temporary remedy exists, such as a server restart. Save the log files before a restart.

Step 2: Review JVM runtime logs

Usually when a login problem occurs, the system log files give some indication. First, look for exceptions and stacktraces. The exceptions should give some clues where to look for the problem. The stacktrace can give information of detailed tracing next.

Step 3: Review the configuration changes if any

If the configuration has some simple change, revert the change and see whether it helps. Sometimes, multiple changes might have all contributed to the problem. Try to revert the changes one at a time to see whether the problem is gone or relieved.

Step 4: Enable traces

If you suspect the login failed during the WebSphere Application Server authentication phase, you may want to add WebSphere Application Server security trace (com.ibm.ws.security.*) to portal trace strings.

One related issue is that multiple persons log in with the same administrator user ID. If these logins are not just for reading or viewing, but try to change some parts of the configuration, it is not supported and potentially can make undesirable results. If multiple administrators are required, add the individual users into the administrator group.

Slow login

When customers report a problem of slow login, usually they mean the span between the time when they submit their user ID and password, and the time when the first page is rendered. It is beneficial to understand what happens after the user ID and password are submitted.

There are four stages after the user ID and password is submitted: the Portal login, WMM retrieval of group information, PAC runtime decision making, and portlet aggregation and rendering on the front page, called the *login landing page* sometimes. During these phases, many components are involved and any one of them can potentially become a bottleneck. To isolate the problem, it is normally an elimination process.

Review the JVM runtime SystemOut.log file to find the suspected spots by inspecting the timestamps of the log entries. Based on this analysis, we recommend a set of traces to be enabled on WebSphere Application Server and WebSphere Portal. To discover whether there is a problem associated with the LDAP server, authentication, or group retrieval, Portal PUMA and WMM traces are recommended. If WebSphere Application Server authentication is suspected, then a WebSphere Application Server security trace should be enabled.

If the bottleneck seems to be outside of portal system, we also suggest the traces to be enabled on other components, such as LDAP, HTTP server, and External Security Manager (ESM), such as Tivoli Access Manager (TAM). In some extreme cases, IP trace may be required.

Figure 4-8 showed the “Login” process from user’s experience.

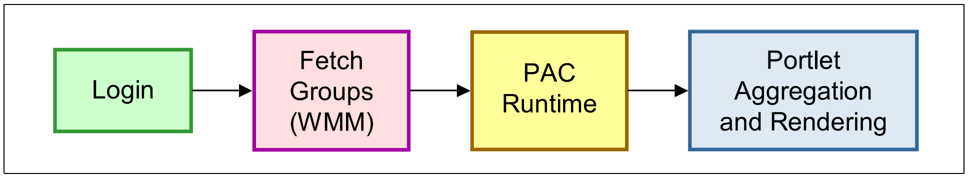


Figure 4-8 General process of “Login”

In the following sections, we give some sample log entries that would identify the start or end of some of the components.

Login phase

In the authentication phase, under normal conditions, the LDAP request should come back fairly fast, that is, in the milli-second range. This can be seen by comparing the timestamps between when we see the Login portlet receiving the user ID and password, and when the user’s full DN is sent back to WMM for the next phase.

Portal login entries usually start with lines shown in Example 4-12.

Example 4-12 Portal login starting point

```
[8/3/07 11:27:54:500 EDT] 0000003f SessionValida >
com.ibm.wps.engine.commands.SessionValidator execute ENTRY [pathData = null,
queryData = { PC_7_NO2UF4I118ADC026HKQ8KC2GT1__login = Log in, password =
********, wps.portlets.userid = wpsadmin }, client = Microsoft Internet Explorer
6.0, locale = en, stateMap = null]
[8/3/07 11:27:54:531 EDT] 0000003f LoginUser     >
com.ibm.wps.engine.commands.LoginUser isAccessToPrivateArea ENTRY [pathData =
null, queryData = { PC_7_NO2UF4I118ADC026HKQ8KC2GT1__login = Log in, password =
********, wps.portlets.userid = wpsadmin }, client = Microsoft Internet Explorer
6.0, locale = en, stateMap = null]
```

```
...
[8/3/07 11:27:54:562 EDT] 0000003f Authenticatio 1
com.ibm.wps.services.authentication.AuthenticationServiceImpl wasAuthentication
(1) new LoginContext
[8/3/07 11:27:54:562 EDT] 0000003f Authenticatio 1
com.ibm.wps.services.authentication.AuthenticationServiceImpl wasAuthentication
(2) lc.login
```

The first entry identified the engine command “SessionValidator” entry point, and the second gave the Login portlet fields “user ID” and “password”. Then the call “lc.login” in “services.authentication” indicates that Portal calls underlie the WebSphere Application Server authentication code. With WMMUR, the WebSphere Application Server authentication code would call the WMM Custom User Registry module to authenticate the user by finding the user and checking the password, and then return the user, as shown Example 4-13.

Example 4-13 Portal login log showing WMMUR authenticates the user “testuser”

```
[8/3/07 11:27:54:656 EDT] 0000003f WMM Trace Log <
com.ibm.ws.wmm.MemberRepositoryManager API: getMember(MemberIdentifier memberId,
StringSet attributeNames, String context) Exit
   memberType:0, memberIdentifier:[uid=wpsadmin,ou=people,ou=dept,o=acme.com /
87d99d40-1f62-102b-8d53-bdbac147b8f0],
parentMemberIdentifier:[ou=people,ou=dept,o=acme.com /
ou=people,ou=dept,o=acme.com]
{sn=sn:Admin, cn=cn:wpsadmin, ibm-primaryEmail=ibm-primaryEmail:wpsadmin@acme.com,
uid=uid:wpsadmin, givenName=givenName:wps, preferredLanguage=preferredLanguage:en}
```

Here WMM returned the DN of the user “wpsadmin” with other attributes. After WMM returned the user, PUMA will then report that the user is logged in after a few checking steps, as shown in Example 4-14.

Example 4-14 Portal log shows a user is logged in and gives the user DN

```
[8/3/07 11:27:54:688 EDT] 0000003f LoginUser     1
com.ibm.wps.engine.commands.LoginUser execute User is logged in:
uid=wpsadmin,ou=people,ou=dept,o=acme.com
```

The entry “User is logged in” indicates that the authentication phase is over.

WMM retrieves group membership information

Next, PUMA calls WMM to retrieve the user’s group membership info, as shown in Example 4-15.

Example 4-15 WMM returned the groups a user belongs to

```
[8/3/07 11:27:54:719 EDT] 00000040 DefaultURMana >
com.ibm.wps.services.puma.DefaultURManager findNestedGroupByUser user=
id: uid=wpsadmin,ou=people,ou=dept,o=acme.com
attributeSubset: [sn, cn, ibm-primaryEmail, uid, givenName, preferredLanguage]
     memberIdentifier: [uid=wpsadmin,ou=people,ou=dept,o=acme.com /
87d99d40-1f62-102b-8d53-bdbac147b8f0]
     attributes: {sn=sn:Admin, cn=cn:wpsadmin,
ibm-primaryEmail=ibm-primaryEmail:wpsadmin@acme.com, uid=uid:wpsadmin,
givenName=givenName:wps, preferredLanguage=preferredLanguage:en}
```

```
     objectID:   [ExtIDImpl
'9eAe0PD8MS4743D0JM466JD4JM46GHC4MM074BD6JM8C4J02MH56KPD46S0CG1'
[87d99d40-1f62-102b-8d53-bdbac147b8f0 / USER, Domain: [Domain: rel]]]
     descriptor: com.ibm.wps.datastore.impl.PrincipalDescriptorImpl@1c8717ba
   objectID: [ExtIDImpl
'9eAe0PD8MS4743D0JM466JD4JM46GHC4MM074BD6JM8C4J02MH56KPD46S0CG1'
[87d99d40-1f62-102b-8d53-bdbac147b8f0 / USER, Domain: [Domain: rel]]]
   created: 1174328690766
   lastModified: 1186154874672
   distinguishedName: uid=wpsadmin,ou=people,ou=dept,o=acme.com
   resourceType: USER
   hasLoggedOut: true
   lastLoginTime: 1186154874672
   markupData: {}
          stack: com.ibm.wps.puma.User
 ENTRY
[8/3/07 11:27:54:719 EDT] 00000040 WMM Trace Log >
com.ibm.ws.wmm.MemberRepositoryManager API: MemberSet
getGroupsForMember(MemberIdentifier memberId, MemberIdentifier baseId, GroupScope
scope, StringSet attributeNames) Entry
                                 [uid=wpsadmin,ou=people,ou=dept,o=acme.com /
null]
                                 <null>
                                 [groupMembershipScope: 1,
membershipHierachyScope: 0]
                                 [cn]
```

When WMM returns, a set of groups would be associated with that user, as shown in Example 4-16.

Example 4-16   WMM returns the group to which the user belongs

```
[8/3/07 11:27:54:750 EDT] 00000040 WMM Trace Log <
com.ibm.ws.wmm.MemberRepositoryManager API: MemberSet
getGroupsForMember(MemberIdentifier memberId, MemberIdentifier baseId, GroupScope
scope, StringSet attributeNames) Exit
[memberType:1, memberIdentifier:[cn=wpsadmins,ou=Groups,ou=dept,o=acme.com /
ca02dec0-1f62-102b-8d55-bdbac147b8f0], parentMemberIdentifier:null
{cn=cn:wpsadmins}, memberType:1,
memberIdentifier:[cn=wcmadmins,ou=Groups,ou=dept,o=acme.com /
b0628340-6840-102b-976f-c7b251c1adc0], parentMemberIdentifier:null
{cn=cn:wcmadmins}]
```

Here “wpsadmins” is returned, which is the group user “wpsadmin” belongs to. When you see “com.ibm.wps.engine.commands.SessionValidator execute RETURN” is printed in the log, you are sure then that the login process should be over and the process of portlet aggregation and rendering starts.

PAC runtime and portlet rendering phase

The permissions of the user is then checked by the PAC runtime to determine the access levels the user has on the login landing page and then the portlets on the page are rendered based on the permissions. To analyze the time spent in this phase, We recommend enabling PAC and portlet traces.

If you suspect the page rendering is the bottleneck, try to eliminate the portlets on the page one at a time to find the most time consuming ones, and move them to the secondary pages. The design of the welcome page should be kept as simple as possible to avoid retrieving data from any back-end systems. One common test is to create a blank page to put in place as the login landing page, thus compare the login landing page with this blank page to discover whether the portlet rendering is the bottleneck. If the portlets on the login landing page require a proxy server to access the internet, forgetting to configure the proxy server may result in a long delay, depending on the timeout of the proxy settings defined by the portlets.

Common causes of slow login

The following list is not meant to be exclusive and there are potentially many more problems that can result in slow login. Sometimes an IP level trace is required to find the bottleneck:

- Slow LDAP server, or database, or network, including faulty network interface card (NIC)
- LDAP referrals
- Firewall caused stale connections
- Nested groups and large number of groups users belong to
- Slow portlet rendering due to back-end systems
- Portlet rendering or proxy access
- Complicated permission settings

If the slow login only happens at the first login right after a server restart, using the “Warmup” service may help. To enable AccessControlWarmupService, locate “WP AccessControlWarmupService” from the WebSphere Application Server Administrative console, and add a custom property with “enabled” as the name and “true” as the value.

You may also want to check the sizes of the Access Control caches, which can be found in “WP CacheManagertService” in the admin console, and follow the suggestions provided in the white paper, *Performance Tuning of Portal Access Control*, found at:

http://www.ibm.com/developerworks/websphere/library/techarticles/0508_buehler/0508_buehler.html

Although this paper was written for WebSphere Portal Version 5, the principles are still applicable to Version 6 as well.

Single sign-on (SSO) is not working

There are several major scenarios in which SSO fails to work.

When the problem is only with the portal security configuration itself, the observation is that after you enter the user ID and password, you are not redirected to the next page; instead, you stay on the same page as though nothing happened. When this happens, make sure that:

- The single sign on domain in global security configuration is configured.
- The fully qualified server host name is used when accessing the portal URL.
- The browser is configured to accept cookies.
- The system timers of the server and client are synchronized.

When the security of the portal server is configured to use an external security manager (ESM) like Tivoli Access Manager, the observation of this problem is that the users have to enter their credentials twice, one with the reverse proxy, and the other at the portal server. As we have emphasized before, you should always make sure the base SSO configuration works before moving on to configure ESM for authentication. You should also check the Trust

Association Interceptor configuration. Further investigation should be done with the traces enabled, using the trace strings given in Table 4-5 on page 107.

The more complicated cases are from the failure of multiple servers. Besides the things mentioned above, you may want to verify the following:

- All participating servers share the same DNS domain, which should be the one configured as the SSO domain. As stated before, the SSO domain cannot be blank in this case.
- All participating servers share the same LTPA key.
- All participating servers are configured to the same user registry and port number, for example, LDAP.

A couple of cases are given in Example 4-17 and Example 4-18.

Example 4-17 SSO failure case: mismatched realm

```
[6/12/07 11:16:37:762 CDT] 0000004d LTPAServerObj E   SECJ0375E: Mismatch of
realms during token validation.
[6/12/07 11:16:37:824 CDT] 0000004d LTPAServerObj E   SECJ0373E: Cannot create
credential for the user <null> due to failed validation of the LTPA token. The
exception is com.ibm.websphere.security.CustomRegistryException: The realm in the
token: tamdirprod.mayo.edu:389 does not match the current realm: WMMRealm

[6/12/07 11:17:03:153 CDT] 0000004d SecurityColla A   SECJ0053E: Authorization
failed for WMMRealm/m024534 while invoking (Bean)ejb/MemberServiceHome
getMember(com.ibm.websphere.wmm.datatype.MemberIdentifier,com.ibm.websphere.wmm.da
tatype.StringSet):1 securityName: WMMRealm/testuser1;accessID:
user:WMMRealm/uid=testuser1,ou=people,ou=dept,o=acme.com is not granted any of the
required roles: Everyone
```

This failure is due to the mismatched user registry realm. When WMMUR is configured, the default realm is “WMMRealm”. If other systems are configured to use the realm, such as “corpldap.acem.com:389”, the configuration in the global security of WebSphere Application Server must be configured to use the same realm. In the case of WMMUR, you need to add a custom property called userRegistryRealm and give the value to the shared user registry realm. This is shown in Example 4-4 on page 111.

Example 4-18 SSO failure case: BadPaddingException

```
[8/13/07 11:12:48:127 CDT] 00000097 LTPACrypto    3
BadPaddingException validating token, normal when token generated from other
factory.
      Given final block not properly padded
[8/13/07 11:12:48:127 CDT] 00000097 LTPACrypto    3   Total decryption time: 1
[8/13/07 11:12:48:127 CDT] 00000097 LTPAServerObj 3   Calling
tokenFactory[2].validateTokenBytes()
[8/13/07 11:12:48:127 CDT] 00000097 AuthzPropToke >  AuthzPropToken from byte[]
Entry
[8/13/07 11:12:48:129 CDT] 00000097 AuthzPropToke 3   Before parsing, length: 169
      string: B4> l<jEQ hV OrgkOE3l?
s <i.CXq] r% E{  w ??# #   H Sg)5"d ]p'B> Y e(Vq  & $Z {O ?_/K1W? ·[[\?] D k
&ySOP3[K]c?j!X?g1ØL!) ym N. 8%"EwY id  ^? ?#kE(@gh 1Pp2;? VCtH)  Tnm _j
[8/13/07 11:12:48:130 CDT] 00000097 AuthzPropToke 3   UserData delimiter not
found.
[8/13/07 11:12:48:130 CDT] 00000097 LTPAServerObj 3
security.ltpa.validate.verifytoken.failed
```

BadPaddingException occurred in this case, and is due to different LTPA keys being used to generate the LTPA token; the failing server could not decrypt the LTPA token.

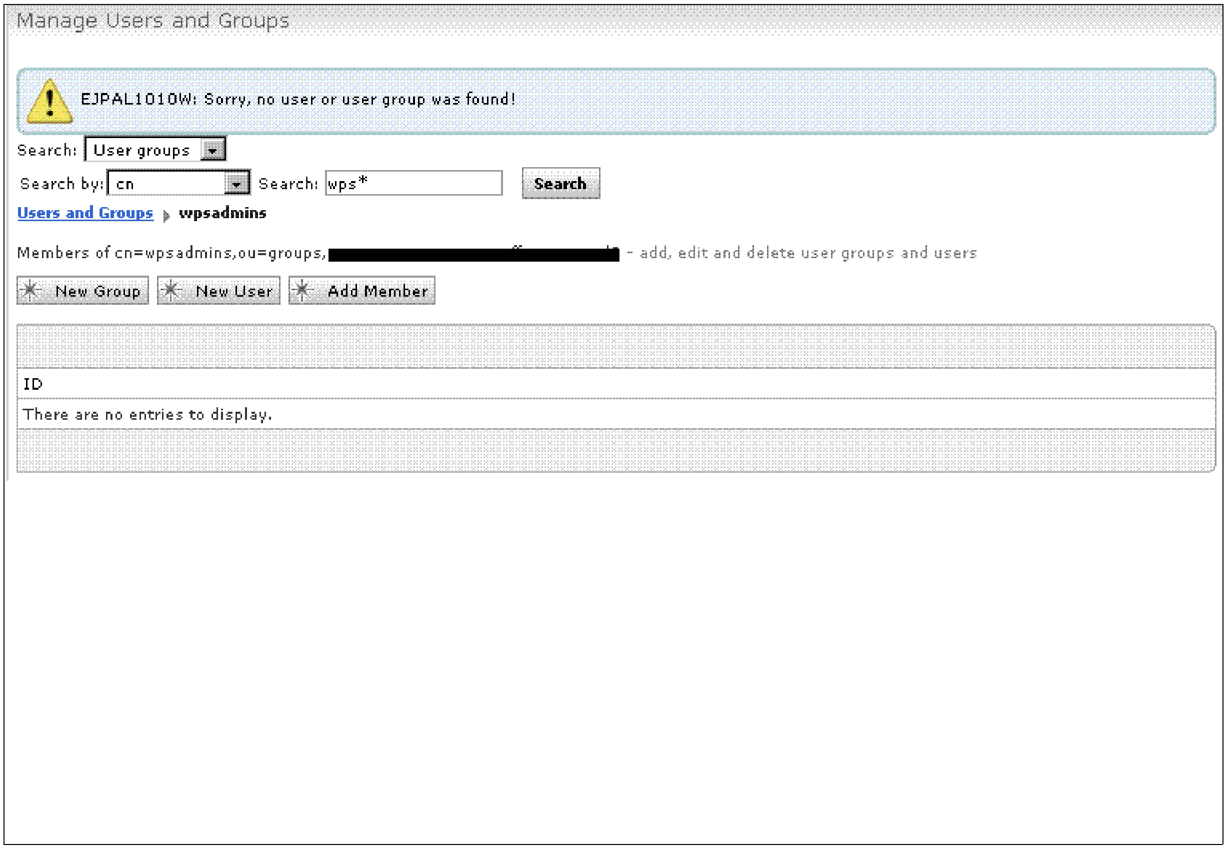
Problems in search of users or groups

The Manage Users and Groups portlet plays important roles in validating and assigning Portal Access Control permissions. It is the tool for portal administrators to manage the users and groups, and manipulates the membership structure without directly accessing the back-end user registry.

After the security is enabled and users are able to log in, they often see problems of locating users or groups, or cannot verify their relationships. There cases most often encountered are:

- Cannot find users or groups, or only some of them can be found, but not others.
- Users are found, but cannot be displayed correctly.
- Not able to find users in a group when clicking the group.
- The group is too large, the sizelimit is exceeded, or a timeout occurs (customize the portlet).

In Figure 4-9, we show a problem of searching groups.



Manage Users and Groups

EJPAL1010W: Sorry, no user or user group was found!

Search: User groups

Search by: cn Search: wps*

Users and Groups ▸ wpsadmins

Members of cn=wpsadmins,ou=groups,[redacted] - add, edit and delete user groups and users

ID
There are no entries to display.

Figure 4-9 No groups found

If you have trouble finding either users or groups, use an LDAP tool to verify that the settings in the WMM configuration is correct. When WMM issues search requests to the LDAP server, it generates the search filter to use the parameters “wmmSecurityAttributeName”, “objectClassForRead”, and “SearchFilter” in wmm.xml.

For example, assume that you search on “john*” on attribute “uid”, and have the WMM configuration shown in Example 4-19.

Example 4-19 WMM LDAP entry configuration

```
   <supportedLdapEntryTypes>
      <supportedLdapEntryType name="Person"
         rdnAttrTypes="uid"
         objectClassesForRead="inetOrgPerson"
         objectClassesForWrite="inetOrgPerson"
         searchBases="ou=people,ou=dept,o=acme.com"/>
      <supportedLdapEntryType name="Group"
         rdnAttrTypes="cn"
         objectClassesForRead="groupOfUniqueNames"
         objectClassesForWrite="groupOfUniqueNames"
         searchBases="ou=groups,ou=dept,o=acme.com"/>
......
   </supportedLdapEntryTypes>
```

The search filter sent to the LDAP by WMM would look like (&(uid=john*)(objectclass=inetorgperson)) with a search base of “ou=people,ou=dept,o=acme.com”.

Using an LDAP utility such as **ldapsearch**, we issue the following command to verify the same configuration:

```
ldapsearch -h corpldap.acme.com -p 389 -b "ou=people,ou=dept,o=acme.com" -D
<bindDN> -w <password> "(&(uid=john*)(objectclass=inetorgperson))"
```

where <bindDN> is the bind user used in WMM configuration, and <password> is the password for the bind user.

If you are able to search for users or groups by attributes, but there is a problem of finding their membership information, such as a failure to find the groups a user belongs to, or the users in a group, then the problem likely resides in the configuration of group to member relationships. The first step is to check the user to group membership mapping.

Without realm support, you should check the setting in “group member ID map” of the advanced LDAP configuration in WebSphere Application Server global security. There are two ways to specify the user to group relationship in the field:

- Multiple “objectclass:property” pairs separated by semicolons. In an objectclass:property pair, the object class value is the same object class that is defined in the group filter, and the property is the member attribute. The examples are “groupOfUniqueNames:uniqueMember” and “groupOfNames:member”. Note that “uniqueMember” always goes with “groupOfUniqueNames”, and “member” with “groupOfNames”. Never mix them.
- Multiple “group attribute:member attribute” pairs separated by semicolons. For some LDAP servers, such as IBM Tivoli Directory Server and Microsoft Active Directory, a user entry is automatically assigned an implicit “group attribute” in which all groups the user belongs to would be stored. Its purpose is to improve performance when you search the groups of a user. Without such an attribute, the search has to exhaust all the groups within

the search base to verify whether the user is in every one of them. When configuring WebSphere Application Server security, you can take advantage of this feature if the underlying LDAP has such an attribute. For example, in the case of IBM Tivoli Directory Server, you can specify “ibm-allGroups:uniqueMember;ibm-allGroups:member”. In the case of Microsoft Active Directory, you can specify “mmeberOf:member”.

In WebSphere Application Server V6.0.2.13 or later, a different baseDN can be specified for group search. You can add a custom property under the LDAP Advanced settings:

```
com.ibm.websphere.security.ldap.groupBaseDn
```

This should give you narrower search space in order to locate the groups faster.

When the security is enabled with realm support, then the search of users and groups is entirely controlled by WMM configurations. So you can set similar configurations in wmm.xml. The configuration fields are:

- groupMemberAttributeMap: This is similar to the first case in "group member id map". A example is “groupOfUniqueMembers:uniqueMember”.
- groupMembershipAttributeMap: This is similar to the second case for the group membership attribute defined in user record. Examples are “ibm-allGroups:uniqueMember” for IBM Tivoli Directory Server and “memberOf:member” for Microsoft Active Directory.
- groupDynamicMemberAttributeMap: WMM added this parameter for dynamic group support. An example is “groupOfURLs:memberURL”.

Another common cause of the search problem is SizeLimitExceededException. In wmm.xml, a default maxSearchResults is defined to be 200. You can manually change the value by directly editing the file, if you anticipate that the returned results will be larger. However, if the sizelimit has to be set very large to accommodate the search result, you may want to consider refining the search or redesigning the LDAP structure.

To debug the problems in searching users or groups, it is always a good idea to generate the LDIF of the branch of the LDAP tree to verify the users and groups and compare the configurations in WMM.

TAM configuration failed

Most of problems of the integration of WebSphere Portal and TAM occur in the configuration phase. As we stated earlier in this chapter, the configuration tasks are intended to run in fairly general scenarios. If your configuration deviates from those presented in the WebSphere Portal Information Center, you may encounter problems. Usually, manual configurations through the TAM admin utility **`pdadmin`** are likely required.

Check wpconfig.properties

Make sure the entries in the file are entered correctly. The TAM administrator user ID and password must be validated before trying the configuration tasks.

Connection to TAM Policy Server

The task validate-pdamin-connection was designed to verify the connection. This is to make sure the portal server can correctly communicate with the TAM Policy Server. If the TAM runtime was set up on the portal correctly, this task should be successful.

Check ConfigTrace.log

When any of the configuration tasks fail, the first thing to look into is ConfigTrace.log. When you find the message BUILD FAILED, scroll up to find the failure error messages close, which should look like something similar to:

```
run-pdjrte-config-non-zos:
[validateHost] Validating hostname(s) in the following argument: tam.acme.com
     [echo] Command to run is: java com.tivoli.pd.jcfg.PDJrteCfg -action config
-host tam.acme.com -was -cfgfiles_path D:/IBM/WebSphere/AppServer/java/jre
     [java] HOST NOT REACHABLE!
     [java] HPDBF0120E   Could not contact the Tivoli Access Manager policy
server.  Possible causes are:
     [java] Connection refused: connect
     [java] The Policy server is not running.
     [java] The Policy server host name or port number is incorrect.
```

Most of these problems are because there are communication problems with the TAM Policy or Authorization servers.

Check runtime logs for exceptions

After the configuration task is successfully run, if there is any problem with loading the trust association interceptor, the configuration of TAI must be reviewed, including the custom properties, the Login modules, the user ID and passwords, and so on.

The first thing to check is if the TAI is loaded successfully by looking for lines similar to those in Example 4-20.

Example 4-20 TAI is loaded successfully

```
[8/17/07 16:44:35:608 EDT]  2934440 TrustAssociat A SECJ0121I: Trust Association
Init class com.ibm.ws.security.web.WebSealTrustAssociationInterceptor loaded
successfully
[8/17/07 16:44:35:731 EDT]  2934440 TrustAssociat A SECJ0122I: Trust Association
Init Interceptor signature: WebSeal Interceptor Version 1.1
[8/17/07 16:44:35:842 EDT]  2934440 TrustAssociat A SECJ0121I: Trust Association
Init class com.ibm.ws.security.web.TAMTrustAssociationInterceptorPlus loaded
successfully
[8/17/07 16:44:35:844 EDT]  2934440 TrustAssociat A SECJ0122I: Trust Association
Init Interceptor signature: Unspecified
[8/17/07 16:44:35:848 EDT]  2934440 SecurityCompo I SECJ0240I: Security service
initialization completed successfully
```

If you see a stack trace indicating a problem, there must be some indication, as shown in Example 4-21.

Example 4-21 TAM vault adapter initialization failed

```
[8/17/07 16:45:21:276 EDT]  2934440 WebGroup      I SRVE0180I: [WebSphere Portal
Server] [/wps] [Servlet.LOG]: ServiceManager: VaultService
[com.ibm.wps.services.credentialvault.VaultServiceImpl] initializing...
[8/17/07 16:45:23:130 EDT]  2934440 WebGroup      I SRVE0180I: [WebSphere Portal
Server] [/wps] [Servlet.LOG]: ServiceManager: exception initializing service
implementation com.ibm.wps.services.credentialvault.VaultServiceImpl:
com.ibm.wps.services.credentialvault.exceptions.AdapterManagerException:
EJPSK0024E: Vault adapter type AccessManager could not be loaded.
```

```
[8/17/07 16:45:23:294 EDT]  2934440 ServletInstan E SRVE0100E: Did not realize
init() exception thrown by servlet portal: javax.servlet.UnavailableException:
Initialization of one or more services failed.
```

In this case, an expired client certificate caused the system to fail.

If there is any message related to the SSL handshake, you need to check the client certificate created when the TAM runtime was configured on WebSphere Application Server.

Enable traces on TAM

In certain cases, it is desirable to enable the WebSEAL traces along with those in WebSphere Application Server and Portal. To enable the WebSEAL Web traces, run the following commands from the PDAdmin console:

```
pdadmin> server task <webseald-server> trace set pdweb.debug 9 file
path=C:\temp\webseald.trace\pdweb.debug
pdadmin> server task <webseald-server> trace set pdweb.snoop 9 file
path=C:\temp\webseald.trace\pdweb.snoop
```

To disable these traces: run these commands:

```
pdadmin> server task <webseald-server> trace set pdweb.snoop 0
pdadmin> server task <webseald-server> trace set pdweb.debug 0
```

If the problem is with authorization with TAM, we recommend adding a “debug=true” custom attribute to the PDLoginModule in the WebSphere Application Server administrative console (select **Security** → **Global security** → **JAAS Configuration** → **Application Logins** → **Portal_Login** → **JAAS Login Modules** → **com.tivoli.mts.PDLoginModule** → **Custom properties** and add debug as the name and true as the value). This will generate debug information to the SystemOut.log upon logging in similar to Example 4-22.

Example 4-22 PDLoginModule debug output

```
[5/26/07 14:46:02:346 EDT] 13de60b4 SystemOut     O delegate class name:
com.tivoli.mts.PDLoginModule
[5/26/07 14:46:02:346 EDT] 13de60b4 SystemOut     O Using the current thread class
loader
[5/26/07 14:46:02:456 EDT] 13de60b4 SystemOut     O user_dn is null
[5/26/07 14:46:02:687 EDT] 13de60b4 SystemOut     O [PDLoginModule]: added
PDPrincipal
[5/26/07 14:46:02:697 EDT] 13de60b4 SystemOut     O [PDLoginModule]: added
PDCredential
```

To reconfigure TAM configuration, do not simply disable security. The TAM settings have to be manually removed from the Portal configuration before trying to disable security.

Portal access control (PAC)

When debugging PAC related problems, check the following:

- Make sure that the user is indeed in the group (if permissions were assigned to groups). One simple test is to assign the user individually and see if that helps.
- Use the XMLAccess utility to generate an export of the object tree, and follow the tree to check that the roles are assigned.
- If rights should not be given and you cannot discover where they were set, check for the virtual principals of which all users are members.

A lot of PAC related problems are due to the settings in the PAC cache settings. People should understand that the cache settings in a production environment is very different from those in a development environment. In a development or test environment, things are more dynamic and changing. Thus, you should set caches to be smaller and the lifetime to be shorter to see an effect. Once in production, all permission settings are tested and fixed, and should not be changed frequently. Thus, you would want to take advantage of the caches to improve the runtime's performance.

When trying to debug any problem of a PAC configuration, it is desirable to generate an XMLaccess export on the release domain (using exportRelease.xml). This XML file will show all the access control configurations of portal resources.

When enabling traces for PAC, be cautious, as an enormous amount of data is generated in a very short time. The user activities should be kept to the absolute minimum. If you know more specific information you want to trace, you can certainly narrow the scope of the trace. For example, the following trace string:

```
com.ibm.wps.ac.impl.PACGroupManagementServiceImpl
```

will only collect information about the group management cache within PAC.

SSL configuration not working

It is important to know how to back out of the configurations you made in case SSL is not working. Thus, always document the steps carefully. When configuring SSL, make sure you are very clear that in the handshake about which party is the client and which is the server. A network diagram should be drawn to show the components involved.

We recommend configuring Portal without SSL first. This reduces the complexity of the configuration.

Step 1: Review runtime logs

The errors or exceptions to be searched are most likely related to the SSL handshake. The example log entries given in Example 4-23 and Table 4-6 on page 135.

Example 4-23 SSLHandshakeException: example 1

```
[8/13/07 23:28:45:406 EDT] 00000042 ManagerAdmin  E
com.ibm.wps.logging.ManagerAdmin initVars EJPFD0055E: Unable to access
traceService MBean.
com.ibm.websphere.management.exception.ConnectorException: ADMC0053E: The system
cannot create a SOAP connector to connect to host localhost at port 10033 with
SOAP connector security enabled.
...
Caused by: com.ibm.websphere.management.exception.ConnectorNotAvailableException:
[SOAPException: faultCode=SOAP-ENV:Client; msg=Error opening socket:
javax.net.ssl.SSLHandshakeException: java.security.cert.CertificateException:
Certificate not Trusted; targetException=java.lang.IllegalArgumentException: Error
opening socket: javax.net.ssl.SSLHandshakeException:
java.security.cert.CertificateException: Certificate not Trusted]
...
Caused by: [SOAPException: faultCode=SOAP-ENV:Client; msg=Error opening socket:
javax.net.ssl.SSLHandshakeException: java.security.cert.CertificateException:
Certificate not Trusted; targetException=java.lang.IllegalArgumentException: Error
opening socket: javax.net.ssl.SSLHandshakeException:
java.security.cert.CertificateException: Certificate not Trusted]
```

The commonly seen SSL handshake problems are summarized in Table 4-6.

Table 4-6 SSL handshake exceptions

Error returned	possible cause
Bad certificate	The certificate is not signed by a known trusted CA.
Unknown certificate	The certificate is not from a known CA chain.
Certificate expired	The date or time associated with the certificate has passed.
Handshake failure	No common cipher protocols available.
Certificate not trusted	An untrusted self-signed certificate in the client.

Reference the WebSphere Information Center for details about these exceptions and how to resolve them.

Step 2: Verify certificates

Depending on what key or trust files are used and whether mutual SSL is configured, use the appropriate tools to open the files to verify the certificates are indeed in them and they are still valid, that is, not expired. When creating your own self-signed certificates or using the default dummy one in the WebSphere Application Server, make a note of their expiration date. For some sites, it may not require a certificate from a Certificate Authority, but the certificates must be replaced before they are expired.

Step 3: Enable WebSphere Application Server security trace and JSSE trace

To set the JSSE trace, add a custom property with the name “javax.net.debug” and value “true” in the WebSphere Application Server admin console for the JVM running. Before verifying portal server applications, try to test some WebSphere applications, such as `snoop`. This is to make sure the WebSphere Application Server configuration is correct.

Step 4: Review portal configuration

If there is an issue with login or logout redirection, then the redirection settings in ConfigService. should be reviewed. Try to put the default setting back and test it.

Sometimes, the mistake might have been made in changing web.xml of wps.ear. Within a cluster, any changes to the web.xml requires a redeploy of wps.ear.

5

WebSphere Portal runtime and services

In this chapter, we discuss the WebSphere Portal Server V6.0.x runtime architecture and the important components that are involved. We will also discuss optimizing the environment, performance tuning, runtime monitoring, and problem determination in detail.

5.1 Overview

WebSphere Portal Server provides an extensible framework for interacting with enterprise applications, content, people, and processes. As such, WebSphere Portal Server acts as a central access point for content, aggregating and displaying content from different applications, the internet, and enterprise content sources. WebSphere Portal Server's self-service features allow users to personalize and organize their own view of the Portal, to manage their own profiles, and publish and share documents with their colleagues.

Most Portals are accessed through Web browsers, but some are also available through wireless Web browsers, voice systems, and other pervasive devices. WebSphere Portal Server supports a variety of desktop and mobile browsers. Furthermore, WebSphere Portal Server is a part of the IBM Application Framework for e-business and acts as a front end to Service-Oriented Architecture.

5.1.1 Portal runtime architecture

The WebSphere Portal Server architecture can be broken down into three main components, as shown in Figure 5-1:

- The Page Aggregator is the engine of the portal, responsible for determining what pages and portlets a user has access to and for assembling the appropriate page based on the request made. The aggregator has several plug-in points, or filters, with which customers may inject custom processing of portal and portlet content, as is done with the transcoding and translation services.
- The Portlet Container and Services are extensions to the J2EE foundation container and services, providing APIs and services that are more portal-centric, including a portlet container as well as services for collaboration, credential vault for SSO management, portlet data management, and so on. All of the J2EE resources are also available.
- The J2EE engine provides the basic Web application container and associated services.

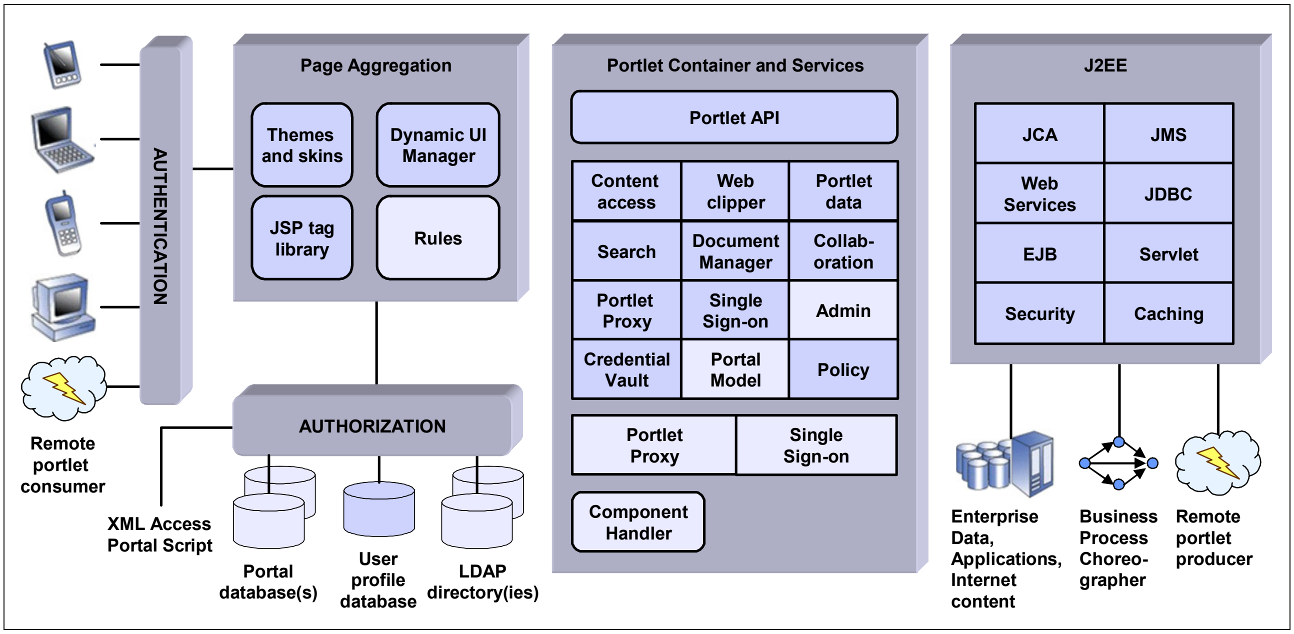


Figure 5-1 WebSphere Portal Server Architecture

In addition, WebSphere Portal Server leverages the foundation capabilities provided by WebSphere Application Server or WebSphere Process Server (certain restrictions apply).

5.1.2 Portal foundation and framework

Although it is usual to refer to WebSphere Portal Server as a single J2EE Enterprise Application, the architecture actually consists of a number of J2EE Enterprise Applications (EAs). It is, however, through the WebSphere Portal application server context, that these EAs cooperate to provide the total Portal experience and application platform.

At the heart of the WebSphere Portal application server is the Portal Servlet, which registers the secure and anonymous context paths (for example, /wps/myportal and /wps/portal, by default). This Servlet receives the browser requests and dispatches specific JSPs to build the Portal page being requested. The JSPs are the same for every page (usually). The layout differs by the configuration for the “page” being requested by the client, as identified by the URI. The actual layout depends on who you are and what access you have to the resources represented by the page’s configuration.

The JSPs do nothing but issue includes for the Portlet Web modules, which are then invoked as servlets, whose service method is translated to the correct render method based on the current mode of the portlet (doView() versus doEdit(), for example).

The following J2EE Enterprise Applications (EAs) constitute the most significant components of WebSphere Portal Server:

- Primary Portal Web application

  The WPS.ear is the primary Portal Web application. It contains the majority of the JSPs responsible for providing the overall Portal look and feel.

- WebSphere Member Manager

  WebSphere Member Manager (WMM) is the component of WebSphere Portal Server that manages data for users and groups. Users and groups are referred to as "members" in WMM. WMM keeps track of the overall attribute set of the users and groups within the system and the values of those attributes for individual users and groups. WMM does not assign particular roles to its members. Members can take on different roles depending on the activities in which they participate.

  Programmatically, WMM presents a Java object view of users and groups to Portal Server, including to all installed Portlets. Details of the actual underlying data storage layouts are abstracted and hidden from Portal Server by WMM. WebSphere Portal Server V6.0.x still uses the Portal User Management Architecture (PUMA) code on top of WMM. This implementation detail should not be important as you begin implementing a custom user store under WMM, but will be important if you need to make use of user profile information in custom-written Portlets or Portal code.

  As the Portal and Portlets make PUMA calls to do searches and get and set attributes on the User object, PUMA passes these function requests to the correct methods on the WMM. WMM then passes the function requests to the correct MemberRepository implementation. Every user managed by WMM requires a unique identifier. A unique identifier allows a member profile to be easily retrieved.

- Personalization Engine

  The Personalization component of WebSphere Portal Server can personalize Portlet content for a specific user. A business user would simply create a rule such as "if the user is a manager, then display management content; if the user is an engineer, then display engineer content."

The JSP views of a portlet can use WebSphere Personalization rules and recommendations in the same way that any JSP page does. This allows the content within the portlet to be personalized, based on the rules and recommendations. Rule and recommendations can also be used in the layout JSP templates or in the page customizer JSP to provide more advanced personalization of the portal.

WebSphere Personalization provides a Personalization Workspace for the business user. This workspace is a browser based tooling for managing rules and campaigns and previewing their effect on your Web site. There are also Development Wizards within WebSphere Studio that create resource classes and content spots.

WebSphere Personalization works by first identifying the site visitor, and retrieving their profile. The Rule/Recommendation engine selects content that matches the user's interest, needs, or role. Finally, the JSP is served to the site visitor.

- Web Content Management

  Web Content Management is a powerful end-to-end Web content management solution that is fully integrated with WebSphere Portal Server. Web Content Management can be used to create and manage Web content for internet, intranet, extranet, and Portal sites. You create the design of Web pages separately from the content of Web pages, allowing different users with different skills to work independently. The structure of your Web site, and the links between the pages in a site, are automatically generated by Web Content Management based on the profile of each content item. You use the access control and workflow features of Web Content Management to control the creation and approval process of Web content.

- Portal Document Manager

  Document Manager helps users organize the content they have seen, want to read, or want to share. Users can create and edit documents without having to be logged in to WebSphere Portal. Users can then upload the documents to Document Manager, which allows other authorized users to work with the documents. Documents are organized into folder hierarchies. Document Manager maintains properties and attributes of documents, handles conversion of documents to other formats, and serves as an organized repository for documents of any format. You can search documents that are maintained in Document Manager by using the Document Manager search feature. Authorized users can control document modifications through versioning and a draft approval process.

5.1.3 Portal Services

The WebSphere Portal Server architecture was designed as a framework and a collection of Portal Services. Many of these Services are implemented as Portlets, but others provide common functionality to Portlets. Each Portlet Service has its own service-specific interface for the functionality that it offers. WebSphere Portal Server supports Portlet Services, including a Portlet container, for both IBM Portlets (now depreciated but still supported) and JSR 168 Portlets. Customers and IBM Business Partners can write their own Portlet Service and register it in the Portal, so that all Portlets can use it. For example, a vendor or customer could write a Search Service, Location Service, or a Mail Service.

The following Portal Services are just some that are provided by IBM with WebSphere Portal Server:

- Administrator Unique Names Mapping Service provides a mechanism for creating URL links between administration portlets, themes, and administrative pages. If these links were hardcoded, they would no longer be usable if you changed the unique names of these pages. Therefore, a service for obtaining those unique names is provided in the AdminUniqueNamesMappingService. This file contains key-value pairs mapping internal keys to the actual unique names, which are assigned to the referenced pages.

- Cache Manager Service is responsible for managing the different caches used in WebSphere Portal Version 6.0.x. The portal provides two different types of caches: shared and non-shared. The shared caches are cluster aware. This means that deleting an element from the cache on one cluster node results in deleting that element from the corresponding cache instances on all other nodes. This ensures that frequently changing data is kept consistent over the whole cluster installation. The non-shared caches are used for data where cluster awareness is of no concern. This avoids unnecessary network communication overhead.
- Collaborative Services allow developers who are writing portlets for WebSphere Portal or other application servers to add Lotus collaborative functionality to their portlets.
- Configuration Service is responsible for collecting the most essential configuration data of the portal engine. Many of these parameters are set by the installation procedure. Therefore, plan well ahead and apply special care when modifying these parameters. The Configuration Service also holds the configuration properties for WSRP services. They are listed and described under Using WSRP services with your portal in the respective topics for which they are relevant.
- ContentAccessService allows portlets to access remote systems or content from remote URLs, including URLs located on the other side of a proxy server.
- CredentialVaultService allows portlets to store and retrieve user credentials for their particular associated application and use those credentials to log in on behalf of the user.
- Data Store Service is responsible for managing the data source of the portal as configured while installing WebSphere Portal Server. Normally there should not be a need to modify any of the configuration parameters in the DataStore service. One important property of the DataStore service is the scheduler.cleanup property.
- Deployment Service provides services for accessing the configuration parameters required for the portlet deployment. The portlet deployment component is responsible for the integration of portlets into the portal. It handles the correct deployment of portlet applications and their WAR files into WebSphere Portal and WebSphere Application Server. It uses the WebSphere Application Server management services for the physical deployment and management of war files in the WebSphere Application Server. Management of war files includes installing, removing, redeploying, starting, and stopping portlet applications.
- Loader Service is responsible for dynamically loading class files in four categories: commands, and supporting classes for screen templates, skin templates, and theme templates. The service does so by looking up a given (class) name in different packages. Upon loading the respective class file, an instance of that class is returned. To optimize the efficiency, the implementation of the service is free to cache loaded class files or instances and return a cached instance. That means that the implementation of any such classes must be thread-safe.
- Navigator Service allows you to specify a number of settings. Among these are settings for cache scope and cache expiration. Depending on your configuration, you might be able improve your performance by modifying these settings. For detailed information about page caching for improved performance, refer to 5.2, “Optimization” on page 142.
- PropertyBrokerService allows Portlets to cooperate and exchange data even if they were developed independently, without the programmer's awareness of the existence of the other cooperative portlets.

- Registry Service loads and caches a small number of objects that are regularly accessed in the engine. This improves performance; however, the trade off is that the cached objects are possibly stale compared to their database counterparts. This applies particularly in a cluster environment. If the age of those objects causes a problem, try reducing the refresh rate for the respective entities.

For a complete list of the Portal Services provided with WebSphere Portal Server V6.0.x, and a description of the many parameters that can be tuned with regards to each service, consult the WebSphere Portal Server Version 6.0 Information Center.

To provide performance and deployment options, some Portal Services, such as search and document conversion, can be executed remotely. The Portal Search service can now reside on a separate machine or as part of a Portal cluster. The remote search service can be utilized either through the SOAP protocol or through EJB. This allows Portal users to search Portal content with the remote search service, but it also allows other search engines to tap into the remote search engine to search portal content. Document conversion is another service that can be deployed remotely. Document Conversion Services is integrated with Portal Document Manager and IBM Common Mail portlet. Documents received as attachments to e-mail can be viewed in the browser even if the application that created the document is not installed. Document Conversion Services also allows documents to be searched by content. Document Conversion Services supports many of the most widely used applications and converts them to HTML and XML. To better balance processing power, Document Conversion Services can be delegated to a remote server. In this case, the service is accessed simply with HTTP, rather than SOAP or EJB. Since WebSphere Portal Server was designed with a Services-Oriented Architecture in mind, we have been able to continually add new services, which can be used by IBM, our customers and our partners to build more powerful Portlets and Portal applications. When customer requirements demand that the service be available outside the Portal framework, we can leverage WebSphere Application Server's support for J2EE and Web Services standards to execute remote services.

5.2 Optimization

Performance is a critical part of any WebSphere Portal Server based solution. In this section, we look at the major parameters responsible for performance tuning WebSphere Portal Server.

> **Important:** No amount of tuning will resolve a performance problem attributed to a badly written Java application. Many customers neglect this important fact and waste a considerable amount of time and effort tweaking around with the WebSphere Portal Server parameters in the hope that a configurational change may resolve matters. Remember, the 20:80 rule: Only a 20% improvement can normally be gained from tuning improvements, while roughly an 80% gain is attributed to development. Therefore, we strongly recommend following the IBM Recommended Best Practices during development.

5.2.1 Knowing where to start

Those familiar with WebSphere Portal Server will be aware that the product is actually deployed as a J2EE Enterprise Application on top of WebSphere Application Server (or WebSphere Process Server with certain restrictions). It is for this very reason that the recommended starting point for implementing a WebSphere Portal Server performance tuning exercise actually involves tuning the underlying WebSphere Application Server instance.

For more information about WebSphere Portal Server performance tuning and a detailed explanation of the parameters, refer to the official *IBM WebSphere Portal Version 6.0 Tuning Guide*, found at:

http://www-1.ibm.com/support/docview.wss?uid=swg27008511&aid=1

Alternatively, refer to the WebSphere Application Server InfoCenter for WebSphere Application Server tuning:

http://publib.boulder.ibm.com/infocenter/wasinfo/v6r0/index.jsp?topic=/com.ibm.websphere.nd.doc/info/ae/ae/welc6toptuning.html

5.2.2 Tuning advice for the IBM Java Virtual Machine

At the core of the WebSphere Application Server is the Java Virtual Machine (JVM). Depending on your selected platform, you will either be running with an IBM JVM implementation, or if on the Sun Solaris platform, the Sun JVM. For functionality and consistency across the WebSphere range, the JVM is installed as part of the WebSphere Application Server package. No dependency is therefore placed on the version of the JVM installed as part of the operating system (usually referred to as the system JVM or default JVM).

JVM heap sizes - IBM JVMs only

One common misnomer is that setting a large JVM heap size improves performance. This is simply not the case. We strongly advise that you choose your Java maximum heap settings carefully and then only based on a thorough Java garbage collection (GC) analysis.

Remember:

- If you use a big heap, then garbage collection will be less frequent but much slower, as there is more memory to search through.
- If you use a small heap, then garbage collection will be more frequent but very fast, as there is less memory to search through.

With the IBM JVM, the Java heap is preallocated (in terms of native memory) at the maximum heap size, reducing the overall amount of memory available to the system.

The Java garbage collection (GC) cycle, which is a stop-the-world (STW) implementation, will prevent the application server from handling loads for a short period of time. All threads are effectively suspended, with the exception of the garbage collection threads, while GC completes to protect the Java heap from corruption. WebSphere vertical clustering can be used to ensure that the CPU is able to provide execution time for at least one cluster member server that can handle load. The IBM JVM supports multiple garbage collection (GC) helper threads to improve performance during the mark phase of GC.

To view or modify the IBM JVM settings from the WebSphere Application Server Administrative Console, select **Servers** → **Application Servers** → **WebSphere_Portal** → **Server Infrastructure** → **Java and Process Management** → **Process Definition** → **Java Virtual Machine**. The default and recommended values are shown in Table 5-1.

Table 5-1 IBM JVM settings

Parameter	Default value	Recommended value
JVM minimum starting heap size	n/a	768
JVM maximum heap size	n/a	1792

Setting the JVM heap size to or greater than 1 GB on AIX necessitates reducing maxdata (the boundary between the permissible data area and the shared memory region in the AIX memory model). However, this shift effectively steals segments from the data area. One limitation that a smaller data area might impose would be in the ability to create native threads, as each thread has a native stack that is in turn allocated from the native heap.

Note that in addition, when using a Java heap greater than 1 GB with the 1.4.x IBM JVM on AIX, the Java heap will be allocated with mmap() and not malloc().

Setting the JVM starting minimum heap size equal to the JVM maximum heap size, without first performing an analysis, gives little overall benefit in terms of long term performance gain. Therefore, we recommend that you refrain from setting the JVM starting minimum heap size equal to the JVM maximum and instead allow the heap to expand until it reaches a steady-state. This will ultimately allow you to determine at what value the heap usage stabilizes. Furthermore, setting the JVM starting minimum heap size equal to the JVM maximum heap size when using a large heap can lead to memory (native heap) fragmentation.

> **Important:** The JVM heap is preallocated (in terms of native memory) at the maximum heap size, so setting a large Java heap size effectively reduces the amount of physical memory available to the system. Under no circumstances should the JVM heap page out to disk. Calculate the free memory available after all other applications and the OS requirements have been taken into account.

Additional IBM JVM attributes

In addition to modifying the JVM heap sizes, the following additional attributes can be evaluated.

To view or modify the IBM JVM settings from the WebSphere Application Server Administrative Console, select **Servers** → **Application Servers** → **WebSphere_Portal** → **Server Infrastructure** → **Java and Process Management** → **Process Definition** → **Java Virtual Machine**. The default and recommended values are shown in Table 5-2.

Table 5-2 Additional IBM JVM settings

Parameter	**Default value**	**Recommended value**
Prevent GC of class loaded data	n/a	-Xnoclassgc
GC Helper Threads	n/a	-Xgcthreads*N*

The -Xnoclassgc setting prevents garbage collection (GC) from reclaiming class loaded data; only de-referenced user objects are reclaimed, and can offer a significant performance improvement. However, the parameter should be fully evaluated for suitability due to the nature of class loaded data. Be aware that dynamically compiled JSPs constitute class loaded data.

With the addition of the -Xgcthreads*N* parameter, the default number of garbage collection (GC) helper threads that are used during the mark phase, can be explicitly overridden. Platforms with *N* processors will have *N*-1 helper threads available by default, which work alongside the master thread to complete the mark phase of GC. *N* is an integer.

If mark stack overflows are observed while performing a verbose garbage collection trace, then additional helper threads may alleviate the problem. Mark Stack Overflows are indicative of deep or wide structures in the Java heap. Processing such structures leads to an overflow of the mark stack, which in turn triggers a sequential heap scan; this is *slow* and consumes

%CPU. More GC threads (-Xgcthreads*N*) will provide more mark stacks (and queues), which means less likelihood of a mark stack overflow. A Java heapdump analysis may also help.

Just-In-Time Compiler (JIT or JITC)

By default, the IBM JVM ships with the JIT (Just-In-Time) compiler enabled. JIT effectively offers a performance improvement by replacing some of the commonly used Java methods and objects with highly optimized C and Assembler routines. This, by necessity, requires an amount of native memory to accommodate the compilation and resulting code set.

It is also worth noting that with JIT enabled, contending methods are not immediately compiled. Instead, such methods are only JIT compiled after they have been invoked after a certain number of times or threshold. This may well present itself as a delayed growth in native memory and could be misinterpreted as a native memory leak (also a performance issue).

It is possible to force JIT to compile methods up front without delay. However, this approach is not normally recommended and thus is not documented here.

Native memory associated with the IBM JVM

It is important to remember that when dealing with a JVM that there is an amount of native memory associated with the process. The types of objects allocated in the native heap or permitted “data area” for the IBM JVM are as follows:

- Just-In-Time (JIT) Compiled code
- Java Native Interface (JNI™) code
- Native Thread Stacks
- Inflators / Deflators
- GZipOutputStreams
- Class Loaded data

IBM JVM CPU utilization

If a system is observed to consume a very high % of CPU for a process associated with the WebSphere JVM, then this could be indicative of spending too much time performing Java garbage collection (GC). In such cases, the recommended action is to use either Tivoli Performance Viewer or a verbose garbage collection trace to identify the characteristics of the Java heap.

If you deduce that it is indeed the Java garbage collection (GC) cycle that is stealing CPU, it does not necessarily indicate a JVM performance or defect issue. More commonly, it may indicate an object management issue in the application or WebSphere. Remember that if a reference to an object is not released, then GC will be unable to free that memory back to the heap; vectors and arrays are particularly prone to this problem if coded incorrectly (session beans and self-grown caches).

5.2.3 Tuning advice for the SUN Microsystems Java Virtual Machine (JVM)

There are major differences between the IBM and SUN Microsystems JVM implementations. It is therefore important to recognize that the SUN JVM includes a generational heap, which splits objects between a Young (Eden) generation (objects are created here) and an Older generation (objects are promoted to the older generation). Each generation is garbage collected (GC) independently and uses a different collection strategy. There also exists a Permanent generation that typically holds class loaded data.

As the Portal Server places a greater demand on JVM memory, you should increase the Java minimum and maximum heap sizes accordingly.

To view or modify the SUN JVM settings from the WebSphere Application Server Administrative Console, select **Servers** → **Application Servers** → **WebSphere_Portal** → **Server Infrastructure** → **Java and Process Management** → **Process Definition** → **Java Virtual Machine**. Table 5-3 shows the default and recommended values.

Table 5-3 SUN JVM settings

Parameter	Default value	Recommended value
JVM initial heap size	n/a	1536
JVM maximum heap size	n/a	2048

Additional SUN JVM attributes

For the SUN JVM, we strongly recommend that you set a number of additional JVM attributes to fine-tune the behavior of the generational heap. Increase the size of the Young generation heap (NewSize and MaxNewSize), as objects in this heap should not live through more then one garbage collection (GC) cycle. Up to 50% to 60% of the total heap size can be set for the Young generation. Also, consider slightly increasing SurvivorRatio.

Additionally, ensure that the Permanent generation (PermSize and MaxPermSize) is adequately sized to hold all class loaded data. This includes classes loaded at Portal Server runtime startup and dynamically compiled JSPs. If the Permanent generation becomes full, a Full GC will result.

> **Note:** The Permanent generation is not allocated from the heap defined by either the (-Xms) or (-Xmx) settings.

To view or modify the SUN JVM settings from the WebSphere Application Server Administrative Console, select **Servers** → **Application Servers** → **WebSphere_Portal** → **Server Infrastructure** → **Java and Process Management** → **Process Definition** → **Java Virtual Machine**. Table 5-4 shows the default and recommended values.

Table 5-4 Additional SUN JVM settings

Parameter	Default value	Recommended value
HotSpot Server option	n/a	-server
Default size of New Generation (in bytes)	n/a	-XX:NewSize=(50% to 60% of heap)
Maximum size of New Generation (in bytes).	n/a	-XX:MaxNewSize=(50% to 60% of heap)
Default size of Permanent Generation	n/a	-XX:PermSize=512m
Maximum Permanent Generation Size	n/a	-XX:MaxPermSize=768 m
Ratio of Eden / Survivor Space Size	n/a	-XX:SurvivorRatio=16
GC Concurrent Collector	n/a	-XX: +UseConcMarkSweep GC
GC Parallel Collector	n/a	-XX:+UseParNewGC

Attention: Incorrectly calculating the various values attributed to the advanced SUN JVM parameters can prevent WebSphere Portal Server from starting up. Always evaluate your parameters in a test or staging environment before undertaking any changes in your production environment.

If you experience performance degradation and high %CPU, consider enabling a verbose garbage collection (GC) trace either through the WebSphere Application Server Administrative Console check box or by using the -verbose:gc parameter. Full GC cycles may be indicative of an insufficiently sized Permanent generation and can give rise to long GC times. Enabling parallel garbage collection may help to reduce such times.

Note: Performance documentation for the various SUN JVMs is available at:

http://java.sun.com/docs/hotspot/index.html

5.2.4 WebSphere resource pools

In order to understand how to maximize performance, it is first necessary to understand the WebSphere queuing mechanism. WebSphere implements a componentized architecture, channeling requests through a number of queues. These queues or pools include the Web server (considered even though it is an external component), the application server Web container, the EJB container, data sources, and possibly other connection pooling mechanisms to various custom back-end systems. Each of these resources sustains a queue of requests waiting to use the resource in question. The overall queuing mechanism is designed to converge towards the back end, where resources are deemed more expensive. For example, it is not uncommon for the Web server queue to be configured to handle an inordinately large number of requests. This contrasts to a data source pool, which by nature is more expensive (both in terms of CPU and memory) and thus usually only configured to handle a maximum of 10-20 connections simultaneously.

Each queue has the potential to become saturated. There also exists the possibility that if one of the back-end queues saturates, that it will have a knock-on effect, impacting the other queues in front. For example, it is not unusual that if a data source connection pool saturates that the Web container will also eventually overload (simply due to the fact that requests cannot be processed further downstream). This can be particularly confusing when instigating a performance appraisal. In which case, it is recommended that you take a holistic approach to performance tuning and determine which queue saturates first.

Tip: One would generally expect to see an increase in %CPU under load when a pool is increased for any given WebSphere Application Server instance. This is the expected behavior, as increasing a resource pool will consume more %CPU as more concurrent throughput is being achieved. However, if the %CPU does not increase after a pool is enlarged when under load, then this usually means there is a bottleneck elsewhere in the system or that the system has reached saturation point.

Consideration should also be taken into account that increasing a resource pool by too much may push a problem to somewhere else within the architecture. For example, a problem relating to SQL query concurrency is only shifted to the database if the concerned WebSphere data source is increased.

5.2.5 Web container

The Web container serves to “gate” the amount of incoming HTTP requests. The larger the number of threads, the higher the number of concurrent requests are allowed to enter the Web container. At some point, however, the number of concurrent threads being processed by the Web container can overwhelm its abilities.

Display Caching should be explicitly enabled for the Portal Web container. In addition, if the WebSphere Dynacache mechanism is going to be utilized by the Portal for Portlet fragment caching, the enablement of Servlet Caching is a prerequisite.

To view or modify the Web container settings from the WebSphere Application Server Administrative Console, select **Servers** → **Application Servers** → **WebSphere_Portal** → **Additional Properties** → **Thread Pools** → **Web Container**. Table 5-5 shows the default and recommended settings.

Table 5-5 Web container settings

Parameter	Default value	Recommended value
Enable Servlet Caching	Disabled	Enabled

In terms of Portal performance, an increase in the maximum number of threads can offer an improvement. However, care should be taken, as increasing the value too high above the suggested optimum value of 75 threads can lead to greater %CPU and memory usage. Setting the number of minimum threads equal to the number of maximum threads does not normally offer any immediate improvement after startup. An examination of a Java thread dump will fail to show a thread count matching the minimum thread setting immediately after initialization.

To view or modify the Web container settings from the WebSphere Application Server Administrative Console, select **Servers** → **Application Servers** → **WebSphere_Portal** → **Additional Properties** → **Thread Pools** → **Web Container**. Table 5-6 shows the default and recommended values.

Table 5-6 Web container settings

Parameter	Default value	Recommended value
Minimum Threads	10	55
Maximum Threads	50	75
Is Growable	Disabled	Disabled

Under no circumstances should the “Allow thread allocation beyond maximum thread size” feature be enabled. Permitting this setting can lead to runaway thread growth. The WebSphere queuing mechanism is designed to handle burst traffic, and occasional “floods” of the Web container should subsume relatively quickly in most circumstances.

Attention: Our experience tells us that one commonly made mistake is that many customers increase the maximum thread pool setting beyond the recommended value in the hope of increasing performance. As this runs the risk of overwhelming the ability of a single JVM, our advice instead is to divide and conquer (D&C) by implementing WebSphere clustering. Running many JVMs, or cluster members, each with a smaller Web container, will prove beneficial when compared to a single JVM deployment with a large Web container.

Custom Web container settings

In addition to the generic parameters just discussed, there exists a number of custom parameters that can be further defined to improve the characteristics of the Web container.

Among these, control of a Web container's Listen Backlog queue is achieve able through the use of the MaxConnectBacklog parameter. If there are more connection requests than available Web container threads, then connections start to backlog, waiting for threads to be freed. If the maximum number of backlog connections is reached, new connections will be refused. Increasing the MaxConnectBacklog queue can extend the number requests queued in the network layer. However, the full implication of increasing the MaxConnectBacklog queue should be understood, as an increased value can effectively lead to latency problems with the WebSphere plug-in resident in the chosen Web server not immediately detecting that an application server has ceased operation (either through a deliberate stoppage or on the occasion of a JVM crash). Choosing the most appropriate value is therefore dependant on the results you wish to achieve.

To view or modify the Web Container Custom Property Settings from the WebSphere Application Server Administrative Console, select **Servers** → **Application Servers** → **WebSphere_Portal** → **Additional Properties** → **Thread Pools** → **Web Container** → **Custom Properties**. Table 5-7 shows the default and recommended values.

Table 5-7 Web container custom property settings

Parameter	Default value	Recommended value
MaxConnectBacklog	512	128
ConnectionIOTimeOut	5	No adjustment
MaxKeepAliveConnections	90%	No adjustment
MaxKeepAliveRequests	100	No adjustment
ConnectionKeepAliveTimeOut	120	No adjustment

The ConnectionIOTimeOut setting can be used to override the maximum time in seconds that a Web container waits when trying to read or write data during a request. It serves to prevent a worst case scenario of runaway connections not timing out. A default value of five seconds is assumed without explicitly setting the parameter.

The remaining three additional custom parameters share the same function as their counterparts found in the httpd.conf configuration of the IBM HTTP Server (IHS).

5.2.6 Data source tuning

The size and behavior for database connection pools managed by WebSphere are maintained by their associated data source configurations. The minimum and maximum pool settings should be set appropriately. Since large pool sizes imply greater resource usage and more contention due to increased concurrency, the best performance is typically achieved with small to moderate connection pool sizes. If performance is at a premium, then the settings for the WebSphere Portal Server data sources should be chosen with care. It is important to understand that increasing the maximum number of connections permitted by a datasource runs the risk of overwhelming the associated database. In terms of performance tuning, an increase can also lead to greater resource usage and more contention due to increased concurrency.

Important: It might at first seem counter-intuitive to refrain from increasing the number of maximum database connections; however, we recommend that a small pool be evaluated first and then, if necessary, increased up to a ceiling of no more than 45 connections maximum. Better performance is generally achieved if the value for the datasource connection pool size is set lower than the value for the Web container connection pool.

Adopting this approach fits well with the paradigm that the WebSphere queuing mechanism is designed to converge towards the back end, where resources are deemed more expensive. Of course, you should ensure that the total number of maximum connections specified across all WebSphere Portal Server cluster members does not exceed the available number connections offered by your selected database. For DB2, this is the MaxAppls parameter, and for Oracle, the number of processes defined in the ora.init file.

The default Prepared Statement Cache setting size on the datasources is generally too small for WebSphere Portal Server. Consider increasing this value, as a larger cache will accommodate more entries and thus prevent useful entries from being discarded to make way for new cache entries. Further analysis can also be determined with Tivoli Performance Viewer, where Prepared Statement Cache discards are an indication of an inadequately sized cache. Caution should nevertheless be exercised, as a larger Prepared Statement Cache will place a greater demand on the Java heap.

A Prepared Statement Cache offers a significant performance improvement to any application that uses the same SQL statement more than once. The same SQL statement implies that the structure of the statement is the same, but that parameter data (the values specified on a where criteria or as update or insert values) can vary. Within WebSphere Application Server, a Prepared Statement is a precompiled SQL statement that is stored in a prepared statement object. This object is used to efficiently execute the given SQL statement multiple times.

Other options relating to connection pooling

For the timeout on requests waiting for new connections, the timeout is currently measured only on the request waiting at the head of the queue, so if the queue is 10 deep, the 10th request will wait for 10 timeout periods before being timed out. Experience suggests that as soon as any queue forms, this policy will result in a rapidly increasing queue length.

The active pool is not shrunk in times of low demand, so we recommend that a shrinkage policy be considered for conservation of system and back-end resources, and also to make the system less prone to connections becoming stale when pooled over long periods of time.

Attention: We strongly recommend that you invest the time and effort in tuning either DB2 or Oracle, as defined in the *IBM WebSphere Portal Version 6.0 Tuning Guide*. For DB2, we found the modifications immediately beneficial, with a Portal response time improvement near 50%. For users of Tivoli Directory Server (TDS), tuning the underlying DB2 database is equally as important.

5.2.7 WebSphere security tuning

When a user logs into WebSphere Portal Server, it is actually the underlying WebSphere Application Server that performs the authentication task (assuming that no authenticating proxy such as Tivoli’s WebSEAL or CA SiteMinder are being used). WebSphere Portal Server then goes on to retrieve the security context from WebSphere Application Server and continues the login. Several WebSphere Application Server security parameters are therefore important when considering the overall performance of the Portal.

Security cache timeout

WebSphere Application Server caches security information related to each authenticated user to save, repeating subsequent User-Registry lookups when a user's security credential expires. This setting controls how long, in seconds, that information is retained before being discarded. As User-Registry lookups ultimately impact performance, we typically recommend that the security cache timeout be increased from the default value. The only exception to this rule might be when modifications to the underlying User-Registry are made, such as invalidating a user after several failed login attempts. In which case, the security cache has the potential to become stale and invalid.

To view or modify the Global Security Settings from the WebSphere Application Server Administrative Console, select **Security** → **Global Security**. Table 5-8 shows the default and recommended values.

Table 5-8 Global security settings

Parameter	Default value	Recommended value
Cache Timeout	600	6000

LTPA settings

Successfully authenticated users receive a Lightweight Third-Party Authentication (LTPA) token containing a credential that can be delegated in the form of an encrypted transient cookie. This cookie is only valid for the duration of a user's browser session and is used through the embedded LTPA token to honor subsequent requests that would otherwise require reauthentication. However, the LTPA token is in itself subject to expiry even if a user's browser session is maintained. Effectively, the LTPA token starts to time out immediately upon creation.

As it is envisaged that users will log in to the Portal at the beginning of the day and maintain a degree of interaction with the system throughout the day, we suggest that the LTPA Timeout be modified to reflect this period. The validity of the LTPA token is also of concern for environments implementing single sign-on (SSO).

To view or modify the LTPA Settings from the WebSphere Application Server Administrative Console, select **Security** → **Global Security** → **Authentication** → **Authentication mechanisms** → **LTPA**. Table 5-9 shows the default and recommended values.

Table 5-9 LTPA settings

Parameter	Default value	Recommended value
LTPA Timeout	120	480[a]
LDAP Search Timeout	120	120
LDAP Reuse Connection	Enabled	Enabled

[a] Dependant on the period of authentication validity required.

One very important parameter with regards to performance and security is the ability to reuse the connection that WebSphere Application Server establishes to the chosen LDAP Directory Server. By default, this parameter "Reuse connection" is enabled.

> **Consideration:** In addition to the LTPA Timeout (absolute), the value defined for the HttpSession Timeout (relative) can impact the behavior of the Portal.

Advanced LDAP filters

We highly recommend that the WebSphere advanced LDAP security filter settings are checked for the most appropriate values according to your chosen LDAP directory server. Failing to corroborate these settings will not only lead to problems with authentication, but can also influence the overall performance of the authentication mechanism.

Two approaches exist for finding LDAP group details. The first uses the GroupFilter to search for groups based on a specified objectclass, for example, groupOfUniqueNames. This is the same approach taken when searching for users with the UserFilter. Unfortunately, this approach scales poorly with large numbers of groups and with large group memberships.

To overcome this issue, many LDAP directory servers now support the listing of groups for which a user is a member as an operational attribute on the actual user object. For example, Active Directory uses the memberOf attribute to hold group membership entries. The correct value should therefore be defined in the Group Member ID Map field. If the attribute is not present, then WebSphere will use the alternative GroupFilter search approach.

To view or modify the Advanced LDAP Settings from the WebSphere Application Server Administrative Console, select **Security** → **Global Security** → **User Registries** → **LDAP** → **Advanced LDAP Settings**. Table 5-10 shows the recommended value.

Table 5-10 Advanced LDAP settings

Parameter	Recommended value
Group Member ID Map	Value from Table 5-15 (See InfoCenter)

Note: The Lotus Domino LDAP implementation only supports the indirect method to locate the group memberships for a user. As such, it is not possible to determine the group membership of a given user by querying the user object directly. Instead, group membership is achieved by iteratively searching through the member list of all groups.

5.2.8 WebSphere session management tuning

User interactions with WebSphere Portal Server are maintained through the use of a HttpSession. This provides a way to preserve data across multiple pages or requests on an individual user basis. It follows therefore that the size of the HttpSession object and the size of the permissible Java heap directly influence the number of users that Portal can concurrently support. Of course, scalability issues can be addressed by WebSphere cloning.

In order to reduce Java heap memory consumption, we typically recommend that the HttpSession timeout setting be reduced from the default value of 30 minutes to 10 minutes. Adopting this approach will then expire the HttpSessions more rapidly, due to the reduced inactivity timeout period and allow Java garbage collection (GC) to eventually reclaim the memory back to the Java heap.

To view or modify the Session Management Settings from the WebSphere Application Server Administrative Console, select **Servers** → **Application Servers** → **WebSphere_Portal** → **Container Settings** → **Web Container Settings** → **Session Management**. Table 5-11 on page 153 shows the default and recommended values.

Table 5-11 Session management settings

Parameter	Default value	Recommended value
Session Timeout (idle time)	30	10

However, the full implication of reducing the HttpSession timeout should be understood. Unlike the LTPAToken timeout setting, which is an absolute timeout value, the HttpSession timeout is based on inactivity and starts to time out each time after a user's last request. If a user fails to interact with the Portal within the timeout period, the session expires and the user will be advised with the message "Your portal session has timed out because of no activity. Start a new session at your portal home."

In most cases, presenting users with this message and a redirection to the WebSphere Portal Server login page maybe acceptable, as Portal navigation is more than intuitive for a user. It is also worthwhile remembering that each Portlet application extends the HttpSession object and has a separate session.

5.2.9 WebSphere Member Manager tuning

The WebSphere Member Manager (WMM) component of WebSphere Portal Server provides an internal mechanism for managing member profiles. Both users and groups are considered members, with WMM constructing a Java object for the resulting entity after performing the necessary interaction with the underlying data store. However, Portal and Portlets do not interface with WMM directly. Instead, requests are handled by the intermediary Portal User Management Architecture (PUMA) layer. This layer of abstraction primarily exists due to historical reasons and is anticipated to be removed in a future Portal version.

The WMM component of WebSphere Portal Server has a number of parameters that can be specifically tuned to improve the login response time and the overall performance of the Portal.

WMM cache configuration

You can tune the way in which WMM caches group information. This can be done by modifying the wmm.xml file and adding the parameters detailed in Table 5-12.

Table 5-12 WMM cache parameters

Parameter	Default value	Recommended value
cacheGroups	false	true
groupsCacheTimeOut	600	3600
attributesCacheSize	2000	2000
attributesCacheTimeOut	600	3600
namesCacheSize	2000	2000
namesCacheTimeOut	300	3600

Important: Since the group cache needs to cache all groups in WMM scope, it may cause a memory problem if there are a large amount of groups. If there are more than 5000 groups in WMM scope, we recommend that groups cache be disabled.

Added support for WMM LDAP connection pooling

By default, WMM creates a single LDAP connection and reuses this connection for all subsequent requests. This is, of course, in addition to the LDAP connection established and reused by the underlying WebSphere Application Server that performs the authentication task on behalf of Portal (assuming that no authenticating proxy such as Tivoli WebSEAL or CA SiteMinder are being used).

Occasionally, several users may simultaneously access the Portal and ultimately the internal WMM component at the same time. For this reason, you can configure WMM to support an LDAP connection pooling mechanism for improved performance. This can be done by modifying the wmm.xml file and adding the parameters detailed in Table 5-13.

Table 5-13 WMM LDAP connection pooling parameters

Parameter	Default value	Recommended value
dirContextTimeToLive	-1	-1
dirContextsMaxSize		10
dirContextsMinSize	1	3
dirContextTimeout	300	3000

Important: Setting the dirContextTimeToLive=-1 means that each connection will be reused forever, until the connection is stale.

Improving group searches

As outlined in "Advanced LDAP filters" on page 152, there are two approaches for finding the group membership for a specific user. Many LDAP directory servers now support listing the groups for which a user is a member as an attribute of the user object (in Active Directory, for example, this is the memberOf attribute). WMM can be configured to use this attribute when asked by WebSphere Portal Server for the groups for which a user is a member, rather than doing an iterative LDAP search for objects of the group objectclass, which have the user DN as a member record. This results in performance improvements for such searches. WMM will still use the group objects themselves when asked to enumerate "all the members of a group". The LDAP directory server itself must be responsible for keeping the attribute in sync with the group member list, so that all groups where the user is listed as a member show up on the attribute, and only groups where the user is listed as a member show up on the attribute.

This can be done by modifying the wmm.xml file and adding the parameter detailed in Table 5-14.

Table 5-14 WMM MemberOf parameter

Parameter	Default value	Recommended value
groupMembershipAttributeMap	n/a	Value from Table 5-15

Table 5-15 on page 155 is a summary of the memberOfAttributeName parameters that various LDAP directory servers support.

Table 5-15 memberOfAttributeName support

LDAP directory server	Value
Active Directory	memberOf
Novell eDirectory[a]	groupMembership
IBM Tivoli Directory Server	ibm-allGroups
Sun ONE Directory Server	nsroles

[a] This attribute is not populated if you add a user to the group through an application other than the Novell eDirectory Administrative Console.

Note: The Lotus Domino LDAP implementation only supports the indirect method to locate the group memberships for a user. As such, it is not possible to determine the group membership of a given user by querying the user object directly. Instead, group membership is achieved by iteratively searching through the member list of all groups.

5.2.10 Portal configuration services tuning

As discussed in 5.1.3, "Portal Services" on page 140, Portal functionality is partially achieved through the deployment of a pluggable framework of services. As such, it is possible to fine-tune a number of parameters associated with each service.

Unlike previous versions of WebSphere Portal Server, Version 6.0.x sees the parameters of each configuration service being set through the WebSphere Application Server Administrative Console. As such, each service is registered as a separate Resource Environment Provider with custom properties that represent the service configuration. You can set these properties by selecting **Resources** → **Resource Environment Providers** → **Browse Clusters** → **WP <Service_Name>**.

Important: It is important to understand that the modification of the Portal Configuration Services should be considered as a specialized fine tuning activity and also as a task, which as a rule, follows on only from first implementing WebSphere Application Server performance tuning.

For more information about the configuration services and their properties, refer to the Portal Configuration Services section of the WebSphere Portal Server Version 6.0 Information Center at:

http://publib.boulder.ibm.com/infocenter/wpdoc/v6r0/topic/com.ibm.wp.ent.doc/wps/srvcfgref.html

Access control data management service

For improved performance during Portal access control lookups, you should avoid using LDAP directories configured with nested groups (a group or groups inside a group). If this is the case and your LDAP directory is not configured with nested groups, then the attributes shown in Table 5-16 can be modified to allow improved performance during user searches by limiting the search to the first level of the group.

Table 5-16 Access control data management service

Parameter	Default value	Recommended value
accessControlDataManagement.enableNestedGroups	true	false

If no nested groups exist in your LDAP or Custom User Registry, the parameter documented in Table 5-16 can be modified. Consult the Information Center for additional parameters that can be modified.

Cache Manager Service

Caching is fundamental to the performance of WebSphere Portal Server. For this reason Portal implements several additional internal caching mechanisms above those found in the underlying WebSphere Application Server instance. For maximum flexibility the characteristics of the majority of these cache instances are configurable through the settings found in the Cache Manager Service. In most environments the default out-of-the-box settings will suffice, leaving modifications only necessary if a high number of cache misses are observed for a concerned entry when viewed with Performance Viewer.

However, one important parameter found under the Cache Manager Service property settings is the cacheglobal.size directive. At first glance this setting would appear to be a catchall for any caches not specified further on in the file.

Table 5-17 Cache Manager Service

Parameter	Default	Recommended
cacheglobal.enabled	true	true.
cacheglobal.size (number of entries)	1000	Increase if necessary.
cacheglobal.shared	false	See notes and InfoCenter.
cacheglobal.replacement	moderate	See notes and InfoCenter.
cacheglobal.admit-threshold	0	See notes and InfoCenter.
cacheglobal.lifetime		See notes and InfoCenter.

Under certain conditions typically associated with aggressive load testing, the global cache can experience thrashing associated with implementing a Least-Recently-Used (LRU) eviction strategy. A worst case load test scenario, for example, might log in 1000 users, at which point in time the global cache will become fully occupied. By increasing the cacheglobal.size value from the 1000 default entries, this problem can be overcome. Careful consideration should nevertheless be exercised, as additional cache entries will consume more Java memory. It also follows that Portal cluster deployments support an accumulative number of entries based on the number of server members participating in the cluster.

Caches may also be shared among all users or maintained on an individual user basis. As this can effect the legitimacy of the caches, we do not recommend modifying the sharing scope of any of the default cache instances.

Clustered Portal environments can on occasion experience cache synchronization issues if the Dynamic Cache Replication Service (DRS) is not implemented.

This typically manifests itself when modifications are made by the Portal Administrator, such as a resource ACL change, with the changes not immediately appearing to be propagated to all the servers participating in the Portal cluster. Under such circumstances, caches are stale and invalid, until they explicitly expire (dependant on each cache's lifetime setting). Restarting the entire cluster or individual cluster members in turn can temporarily overcome this difficulty. However, this should not be considered the fix for the root cause of the problem.

Note: You must enable the DRS in your clustered environment in order to correctly validate the Portal caches. If DRS is not enabled, situations may arise where users have different views or different access control rights (ACLs), depending on which cluster member handles the user's request. It is, however, usual that session affinity is maintained to a specific cluster member for the life of the user's session. The only exception to this is failover. The issue is also dependant on the level of Portal Personalization offered to a user.

As all caches eventually expire, you may accept that a stale cache is an anticipated occurrence and choose to live with the situation. Consult the Information Center for additional parameters that can be modified.

Configuration Service

Several attributes that influence Portal performance are defined under the Configuration Service. Among the most important settings are the persistent session options that offer an authenticated portal user the ability to return to their last visited page from the time of their last session. However, there is a significant impact in enabling this functionality, as the state must be persisted to the Portal database. In most cases, disabling this feature is acceptable, as Portal navigation is more than intuitive for a user. The Configuration Service also holds the configuration properties for Web Services for Remote Portlets (WSRP) services. shows the default and recommended values for the Configuration Service.

Table 5-18 Configuration Service

Parameter	Default value	Recommended value
persistent.session.level	0	May need to be changed. See InfoCenter.
persistent.session.option	0	May need to be changed. See InfoCenter.
timeout.resume.session	false	May need to be changed. See InfoCenter.

Consult the Information Center for additional parameters that can be modified.

Deployment Service

Although not strictly related to Portal performance, the Deployment Service contains several important properties. If Portal is deployed in a cluster, then the was.notification.timeout (in seconds) can be increased to extend the period of time the underlying WebSphere Application Server will wait before timing out from performing the deployment task of any new portlets (worst case scenario). This value may have to be increased for large scale Portal installations. Table 5-19 shows the default and recommended values for the Deployment Service.

Table 5-19 Deployment Service

Parameter	Default value	Recommended value
was.notification.timeout	60	600

Consult the Information Center for additional parameters that can be modified.

Navigator Service

Several attributes found under the Navigator Service can be modified to influence the performance behavior of the Portal anonymous front page. By increasing the public.reload setting, less of an impact will be made on the Portal database, as the database will be queried less frequently to reload the page details. The default value of 60 seconds has the potential to overwhelm the database and can increase without hesitation.

If the Portal anonymous front pages are not likely to change regularly, then the public.expires setting can also be increased. The expiration value can be used by Portal to define the HTTP expires header lifetime for anonymous front pages, in accordance with section 14.9.3 of RFC 2612 The CAST-256 Encryption Algorithm. The expiration value is, however, only applicable for cached responses and not for first-time requests. It effectively sets the duration after which the cached response is considered stale in a user's browser.

Under certain circumstances, it may prove necessary to create a session associated with the Portal anonymous front page. This is achieved by setting the public.session parameter equal to true, but is generally discouraged, as it tends to reduce Portal performance. Table 5-20 shows the default and recommended values for the Navigator Service.

Table 5-20 Navigator Service

Parameter	Default value	Recommended value
public.session	false	false, unless a session is needed
public.reload	60	600 or higher
public.expires	60	60 or higher

Consult the Information Center for additional parameters that can be modified.

Portlet Container Service

Parallel Portlet Rendering (PPR) permits portlets to be rendered in parallel rather than in sequence. This means that the overall page aggregation response time can be greatly accelerated. By default, the Portal is configured for parallel portlet rendering. However, portlets themselves must be enabled to support the parallel rendering feature. Parallel portlet rendering is particularly suited for portlets that call remote services or perform intensive I/O, as a separate thread is spawned for each portlet. Table 5-21 on page 159 shows the default and recommended values for the Portlet Container Service.

Table 5-21 Portlet Container Service

Parameter	Default value	Recommended value
parallel	false	true
legacy.useParallelRendering	false	true
std.useParallelRendering	false	true

Consult the Information Center for additional parameters that can be modified.

PUMA Service

The options configured under the PUMA Service affect the performance characteristics of the internal PUMA layer, the function of which is to build a member object associated with a user's specific attributes. This is achieved in part by submitting a request to another internal Portal component called WWM. For efficiency, PUMA was designed to initially request a minimum subset of attributes from WMM, which would in most circumstances fulfill most member object requests.

The user.base.attributes property is a comma separated list of attributes that will be requested initially from WMM by PUMA when a user first logs in. The user.minimum.attributes property is a comma-separated list of attributes that will be requested initially from WMM by PUMA. If Portal or a Portlet requests an attribute that is not defined in the list, PUMA is then forced to make a subsequent request for the entire attribute subset. This is somewhat costly in terms of performance, as additional queries to the user data store will result. Table 5-22 shows the default and recommended values for the PUMA Service.

Table 5-22 PUMA Service

Parameter	Default value	Recommended value
user.minimum.attributes	uid,cn	See notes.
user.base.attributes	uid,cn,givenName,sn,preferred Language	See notes.
group.minimum.attribute	cn	See notes.

You should ensure that both the user.minimum.attributes and group.minimum.attributes settings contain the attributes deemed necessary for your requirements. If Portal (or a Portlet) requests an attribute that is not present in any of the above lists, PUMA will make a second request to the user registry. However, such a request will actually be for a full attribute set retrieval, from the user registry through WMM.

Registry Service

The parameters found under the Registry Service control the expiry times of cached instances of various Portal objects read from the database, such as cached Portlets, themes and skins, and so on. By default, the majority of these buckets are set to expire at the same regular time interval. However, a semaphore does exist in the applicable code to prevent simultaneous reloading, which would otherwise lead to temporary lockups. For beneficial results, setting the values at slightly different intervals may help to prevent the refreshes from hitting the database at the same time. Furthermore, the staggering can be extended to horizontally clustered Portal Server deployments.

5.3 Problem determination

Dealing with WebSphere Portal Server problems can at first seem a daunting prospect, even to the most accomplished Portal Administrator. However, with a little knowledge and direction, you can quickly become the master of a situation, quickly identifying and rectifying the problem to a successful resolution.

In this section, we endeavour to share with you some of the techniques commonly used and endorsed by the IBM WebSphere Support Team in determining the root cause of problems and solving the problems. Understanding the components involved with WebSphere Portal Server will greatly help your diagnostic and problem solving skills. So, we strongly recommend that you first understand how all the WebSphere Portal Server components work and how all the components are integrated to efficiently debug a problem.

This section of the Redpaper compliments *WebSphere Portal Version 6 Enterprise Scale Deployment Best Practices*, SG24-7387 and the InfoCenter, which have very good information about problem determination and troubleshooting. Refer to:

http://publib.boulder.ibm.com/infocenter/wpdoc/v6r0/index.jsp?topic=/com.ibm.wp.ent.doc/wps/pd_stepone.html
http://www.redbooks.ibm.com/redbooks/pdfs/sg247387.pdf

5.3.1 Identify the failing component

WebSphere Portal Server should be considered as a horizontal framework rather than a sole application because a complete Portal Solution is comprised of many different components. So, with all these components in place, it is very important to narrow down exactly the failing component in case there is a problem.

5.3.2 JVM problems

Understanding JVM is very important because the IBM WebSphere platform is built on Java and the Application Server is a Java-servlet-based application deployment environment for server-side applications and JavaBeans. Let us discuss some common problems with JVM and how we can diagnose them.

When is a crash a crash and not a hang

The properties of a crash and a hang at either level are basically the same. A hang occurs when a process or thread gets stuck waiting for something (usually a lock of some kind or some software/hardware resource) to become free. Waiting for a lock or a resource is not uncommon, but it is when that lock or resource does not become available that a hang occurs.

It is also important to note that hangs can sometimes be diagnosed too early. For example, a resource is very busy at a given time; a process or thread that needs to use that resource may then have to wait an unusually long time for that resource to become free. A user may be unaware that the resource is busy and only sees the process waiting, so he interprets that as a hang when it is actually working as designed, albeit slowly.

A crash is very different from a hang and occurs when an unexpected hardware or software error occurs. When these errors occur, special error handling is hopefully invoked to dump out diagnostic information and reports that will hopefully be useful to track down the cause of the error. Crashes can be thought of as point-in-time problems that require post-mortem analysis, and hangs can be thought of as real-time problems that one can analyze live.

Refer to Appendix A, “Using IBM tools to find solutions and promote customer self-help” on page 169 for the tools available to diagnose crashes and hangs.

JVM crashes

Under normal conditions, Java is supposed to catch exceptions and handle them. So, if the JVM is found to be crashing, then this must be attributed to either JITC code (because it uses native compilation) or JNI code (because, again, it uses native code) and lastly possibly because of the JVM itself. If the JVM crashes, the following aspects should be investigated:

- System environment

  The system might have been in a state that has caused the VM to crash. For example, there could be a resource shortage (such as memory or disk) or a stability problem. The javacore file tells you how to find disk and memory resource information. The system logs can give indications of system problems.

- Java environment

  Use the javacore file to determine what each thread was doing and which Java methods were being executed.Use the -verbose:gc option to look at the state of the Java heap.

JVM signals in UNIX

AIX and Solaris, like other UNIX based operating systems, make use of signals. Signals are, of course, a method by which the operating system can interrupt a running process or program. Some of the common signals that result in abnormal program termination are listed Table 5-23.

Table 5-23 JVM signals

Signal name	Signal type	Description
SIGBUS (7)	Exception	Incorrect access to memory (data misalignment).
SIGSEGV (11)	Exception	Incorrect access to memory (write to inaccessible memory).
SIGILL (4)	Exception	Illegal instruction (attempt to invoke an unknown machine instruction).
SIGFPE (8)	Exception	Floating point exception (divide by zero).
SIGABRT (8)	Error	Abnormal termination. The JVM raises this signal whenever it detects a JVM fault.
SIGINT (2)	Interrupt	Interactive attention (CTRL-C). JVM exits normally.

Using dbx or dbxtrace against the core

dbx is the AIX standard command-line debugger. You can automate dbx into diagnostic "probes". IBM Support might ask you to obtain and run selected probes, either against a test instance of the troubled application, or against the dump files generated by an application failure.

Refer to the IBM Java SDK InfoCenter for more information about using the dbx utility, found at:

http://publib.boulder.ibm.com/infocenter/javasdk/v6r0/index.jsp?topic=/com.ibm.java.doc.diagnostics.60/diag/problem_determination/i5os_dbx_sysdump.html

JVM hangs

Knowing what to do when WebSphere becomes unresponsive will greatly improve your ability to move a problem to a successful resolution. For this reason, we consider the situation when a WebSphere JVM appears to hang on the AIX platform. Over the next few subsections, we will see how it is potentially possible to determine the cause of such a JVM hang.

Without necessarily knowing Java or the low-level specifics of the underlying AIX operating system, you can get straight to work and put these techniques into practice should you experience a hang.

Debugging hangs

If the JVM is hanging and the process is still present but is not responding, then the following causes should be investigated:

- The process has come to a complete halt because of a deadlock condition.

  A lock (also referred to as a monitor) prevents more than one entity from accessing a shared resource. Each object in Java has an associated lock (obtained by using a synchronized block or method). In the case of the JVM, threads compete for various resources in the JVM and locks on Java objects.

  Example 5-1 was taken from a deadlock test program where two threads “DeadLockThread 0” and “DeadLockThread 1” were unsuccessfully attempting to synchronize on two java/lang/Integers.

  You can see in Example 5-3 on page 166 that “DeadLockThread 1” has locked the object instance java/lang/Integer@004B2290. The monitor has been created as a result of a Java code fragment looking like “synchronize(count0)”, and this monitor has “DeadLockThread 1” waiting to get a lock on this same object instance (count0 from the code fragment). Below the highlighted section is another monitor locked by “DeadLockThread 0” that has “DeadLockThread 1” waiting.

  This classic deadlock situation is caused by an error in application design; javadump is a major tool in the detection of such events.

Example 5-1   Deadlock example

```
LOCKS subcomponent dump routine Monitor pool info:
     Current total number of monitors: 2
 Monitor Pool Dump (flat & inflated object-monitors):
      sys_mon_t:0x00039B40 infl_mon_t: 0x00039B80:
       java/lang/Integer@004B22A0/004B22AC: Flat locked by "DeadLockThread 1"
                  (0x41DAB100), entry count 1
            Waiting to enter:
                "DeadLockThread 0" (0x41DAAD00)      sys_mon_t:0x00039B98 infl_mon_t: 0x00039BD8:
       java/lang/Integer@004B2290/004B229C: Flat locked by "DeadLockThread 0"
                                                             (0x41DAAD00), entry count 1
            Waiting to enter:
                "DeadLockThread 1" (0x41DAB100)
JVM System Monitor Dump (registered monitors):
          Thread global lock (0x00034878): <unowned>
          NLS hash table lock (0x00034928): <unowned>
          portLibrary_j9sig_async_monitor lock (0x00034980): <unowned>
          Hook Interface lock (0x000349D8): <unowned>

Deadlock detected !!!

  Thread "DeadLockThread 1" (0x41DAB100)
    is waiting for:
      sys_mon_t:0x00039B98 infl_mon_t: 0x00039BD8:
      java/lang/Integer@004B2290/004B229C:
```

```
    which is owned by:
  Thread "DeadLockThread 0" (0x41DAAD00)
    which is waiting for:
      sys_mon_t:0x00039B40 infl_mon_t: 0x00039B80:
      java/lang/Integer@004B22A0/004B22AC:
    which is owned by:
  Thread "DeadLockThread 1" (0x41DAB100)
```

- The process has become caught in an infinite loop.

  If there is no deadlock between threads, consider other reasons why threads are not carrying out useful work.Usually, this state occurs for one of the following reasons:

  – Threads are in a '"wait" state waiting to be "notified" of work to be done.

  – Threads are in explicit sleep cycles.

  – Threads are in I/O calls waiting to do work.

  The first two reasons imply a fault in the Java code, either that of the application, or that of the standard class files included in the SDK.

  The third reason, where threads are waiting (for example, on sockets) for I/O, requires further investigation. It will be worth checking to see if the process at the other end of the I/O failed or there are any network problems.

- The process is running very slowly.

  If no infinite loop is occurring, look at the process that is working, but having bad performance. In this case, change your focus from what individual threads are doing to what the process as a whole is doing.

Refer to Appendix A, "Using IBM tools to find solutions and promote customer self-help" on page 169 for more information about using tools to analyze hangs and crashes.

5.3.3 Some common problems and workarounds

There is ample information in the above mentioned IBM Redbooks publication and the InfoCenter about the problems and troubleshooting techniques. However, we will discuss some issues we faced in our environment and the solutions.

Failed installations

- On a UNIX machine, after running the **install.sh** command, nothing happens. No logs are updated in the /tmp directory, and the process also does not seem to be running. This machine had a previous failed installation of WebSphere Portal Server.

  The uninstallation steps were not thoroughly followed and we later noticed that the VPD.properties file still existed on the file system and this file had the Portal entries for the previously failed Portal installation.

  Make sure to follow the uninstallation steps completely for each environment.

- Most of the installations fail because there is not enough space in the /tmp space, so be sure that you have enough space before you start the installation.

LDAP security problems

- Most of the enable-security task failures occur because of incorrect parameters, which are often overlooked. Once you have entered all the values for the properties in the configuration file (wpconfig.properties), we recommend that you use an LDAP client and try to do a simple connect to the LDAP server with the parameters that you just entered in the configuration file. That way, we can debug the LDAP server connection problems and see if there are any problems with the "Bind" account permissions.

- Login delay.

  There are a number of components involved with the Portal login, such as the database, LDAP, WMM configurations, and so on. Portal login is explained in Chapter 4, “WebSphere Portal security” on page 85.

  In our environment, we had IBM WebSphere Portal V6 with Active Directory. The login time was a bit long (around 18-20 seconds). We performed a TCP dump and analyzed the dump file and found out that there is a 12 second delay from the LDAP server after a request is sent. We then analyzed the search base for Portal and narrowed the cause down to the user and group objectclasses being used and indexed those object classes in the Active Directory server, which immediately reduced the login time from 15-20 seconds to 2 seconds. However, be aware that the indexing might actually cause some impact on the LDAP server, so before implementing our solution or a similar one, take the LDAP server’s processing power and memory into consideration.

  So, it is not the Portal alone that is entirely responsible for the delay; it is worth looking at the other involved components as well.

5.4 Portal administration tools

There are some very efficient administration tools available to you to automate the deployments, move the configuration across environments, and so on.

You can administer and configure portal resources by using one of the following tools:

Portal administration portlets

Portal administrative users can use the administration portlets to perform administrative tasks and actions on portal resources, depending on the access rights that the administrative user has on those resources. This includes:

- Configuring individual portal resources.
- Configuring individual portal resources, together with their dependent resources. For example, this can be pages and the pages derived from them.
- Giving other users, for example, subadministrators, limited access rights on selected portal resources. These subadministrators can then perform administrative tasks to the extent that their access rights allow. As the master administrator, you can widen or limit that extent by modifying the access rights for these users on the portal resources. This way, you can delegate administrative tasks as required., or even deploy your own custom developed artifacts, such as portlets, themes, or skins.

  You cannot use the administration portlets to perform scripted or automated administration or configuration tasks.

  For more information about the Portal Administration tools, refer to the InfoCenter at:

  http://publib.boulder.ibm.com/infocenter/wpdoc/v6r0/index.jsp?topic=/com.ibm.wp.ent.doc/wps/adpltadm.html

XML configuration interface

The XML configuration interface works as follows

- The XML configuration interface provides a batch processing interface for portal configuration updates. It allows you to export an entire portal configuration or parts of a configuration, for example, specific pages, to an XML file. You can then re-create the exported configuration from such a file on another portal.

- You access the XML configuration interface using a command-line tool. This command-line client is a small separate program that connects to the server using a HTTP/HTTPS connection. You can therefore use it remotely. You can use the XML configuration interface to process portal resources, but not portal actions or tasks.
- You can use the XML configuration interface to process the configuration of portal resources that already exist, for example, pages. In this context, the XML configuration interface processes derived resources, but it does not automatically create them.
- The XML configuration interface does not reflect the access control authorization model with delegated administration. You only need the access permission to use the XML configuration interface. An administrator who works with the XML configuration interface does not need access permission for the portal resources processed by the XML request. (The reason for this is that access control gives users access permissions on actions and not on resources.)

You can use the XML configuration interface for the following tasks:

- Exporting, importing, and updating complete or partial portal installations. This can be for the transfer or migration between machines.
- Backing up the portal configuration. (Remember that XMLAccess is not a solution for the complete Portal environment backup; for a complete Portal environment backup, refer to Appendix B, “Maintenance: Fix strategy, backup strategy, and migration strategy” on page 207.)
- Overview of the portal configuration.
- Cloning of a portal.
- Copying parts of a configuration, such as specific pages, from one portal to another.
- XMLAccess should only be used to transfer the initial content when moving from staging to production. Subsequent differences and updates should be transferred using the ReleaseBuilder.
- Creating a portal configuration file by XML export. You do this by performing an XML export.
- Installing additional resources on a portal.
- Performing recurring administration tasks in an automated and reproducible manner.
- Performing these administrative tasks remotely, that is, from another server through an HTTP connection.

For more information about how to use XML access and understand the complete process, refer to the IBM Redbooks publication *WebSphere Portal Version 6 Enterprise Scale Deployment Best Practices*, SG24-7387, found at:

http://www.redbooks.ibm.com/redbooks/pdfs/sg247387.pdf

Troubleshooting the XML configuration interface

The topic gives information to assist you with preventing, identifying, and correcting possible problems that might occur when using the XML configuration interface.

- Unexpected syntax errors.

  A possible reason is that your files are being truncated before they are processed by the portal. This can typically occur for the following reasons:

  - Your input file contains invalid UTF-8 characters. A good way to check your input files is to view them with Microsoft Internet Explorer. Microsoft Internet Explorer shows errors if your input contains invalid characters.

- Some vi editors of UNIX systems cause problems when handling large files. This depends on the implementation of the vi editor. For example, if you use a vi editor to modify an XML script with more than 40.000 lines, parts of the file contents might get truncated. Use a different editor to modify such large files.
- You have a problem with your HTTP communication setup, for example, your input is relayed through a HTTP server that truncates it. In this case, you can check for communication problems by specifying the -echo command-line parameter when you call the XML configuration interface tool. When this parameter is specified, your request is not processed, but simply returned as it is read by the server. If the output is different from your input file, your request was modified somewhere along the communication path.

- Message "LDAP: error code 49 - Invalid Credentials".

  If you try to connect to the portal with security and LDAP enabled, but if the LDAP server is not available, the error message shown in Example 5-2 is returned in the XML response file.

Example 5-2 LDAP error message

```
com.ibm.websphere.wmm.exception.WMMSystemException:
     The following Naming Exception occurred during processing:
     "javax.naming.AuthenticationException: [LDAP: error code 49 - Invalid
Credentials]".This message can be misleading.
```

Solution: The LDAP error message `Invalid Credentials` means that the user name or password are wrong. It can also mean that the LDAP server is not available at all.

- Message "LDAP: error code 50 - Insufficient access".

  This message is returned when the user does not have proper access to the users. The admin account used to run the task should have at least read access to the LDAP tree.

- Message"XMLC0142E: Unique name unique_name is already used in the portal".

  Solution: When you create a nested element, for example, a component with a uniquename attribute, the whole hierarchy upward from that element must also have uniquename attributes. Example 5-3 shows an example XML export request snippet.

Example 5-3 XML export example

```
<content-node ...
   <component uniquename="component_1"...
         <component uniquename"component_2"...
              <component uniquename"component_3"...
                   . . . . .
               </component>
          </component>
   </component>
</content-node>
```

- Cannot use the XML configuration interface if it is externalized in security.

  If the virtual resource XML_ACCESS that represents access to the XML configuration interface is put under the protection of an external security manager, you can no longer use the XML configuration interface.

  Solution: If the access rights of WebSphere Portal are externalized to an external security manager, such as Tivoli Access Manager, make sure that the XML configuration interface virtual resource is not externalized

For more information about some common problems and solutions with the XML access tool, refer to the WebSphere Portal Server InfoCenter at:

http://publib.boulder.ibm.com/infocenter/wpdoc/v6r0/index.jsp?topic=/com.ibm.wp.ent.doc/wps/adxmltrb.html

WebSphere Portal ReleaseBuilder

During the staging of follow-on releases of IBM WebSphere Portal, portals, configurations, and artifacts need to be moved between systems. ReleaseBuilder enables the management of release configurations independent of user configurations. Release configuration data can be exported into an XML configuration interface configuration file. During the staging of follow-on releases, it is now possible to stage release configurations between two releases using the XML configuration interface. This allows you to track which configuration entities were removed, added, or changed compared to the previous release, and apply differential updates. Differential updates are created by detecting the differences between one configuration and another. A third configuration can then be generated to represent the difference and used to apply not only additional and update modifications, but also deletions to the production server. This allows the staging and production servers to remain in sync and eliminates the problem of configuration bloat on the production server.

ReleaseBuilder is a configuration management tool. It should not be used for migration purposes.

Refer to the WebSphere Portal Server V6 InfoCenter for more information about ReleaseBuilder at:

http://publib.boulder.ibm.com/infocenter/wpdoc/v6r0/index.jsp?topic=/com.ibm.wp.ent.doc/wpf/dep_rbabout.html

Portal Scripting Interface

IBM WebSphere Portal provides a scripting interface that enhances the possibilities for automated solution deployment and administration of the portal. The Portal Scripting Interface allows you to create scripts that portal administrators can use to perform administrative tasks from a command line.

The Portal Scripting Interface allows portal solution development teams to write scripts that are later executed by operation teams for solution deployment. These scripts have the same functionality as the portal administration user interface. However, the scripting interface does not facilitate all operations that can be done using the Portal UI. This allows you to implement automated configuration management for various kinds of configuration changes.

Scripts support the work split between solution development and solution operation teams. Even if the solution development teams cannot work interactively with the production system, they can apply the same administrative actions through the use of scripts. At the same time, the use of scripts enhances availability and quality of the solution, as developers can write and test the scripts without interfering with the production system. Scripts provide repeatability and avoid user errors that are likely in manual administration procedures.

Refer to the WebSphere Portal Server V6 InfoCenter for more information about the scripting interface at:

http://publib.boulder.ibm.com/infocenter/wpdoc/v6r0/index.jsp?topic=/com.ibm.wp.ent.doc/wps/ad_psi.html

5.5 Runtime monitoring

In today's market, there are probably hundreds of monitoring tools, so finding the right tool for your environment is a huge challenge even before you think about "how to monitor". So, choosing the right monitoring tool is definitely the first step in the process. However, this topic is out of the scope of this Redpaper, as it involves many man hours and depends on each company's corporate policies in choosing software.

WebSphere monitoring involves delivering comprehensive fault management and proactive alert notifications, checking for impending problems, triggering appropriate actions, and gathering performance data for planning, analysis, and reporting. There are lot of third-party tools available for monitoring WebSphere Portal and also some IBM tools, such as IBM Tivoli Composite Application Management (ITCAM) and PV(Performance Viewer).

> **Important:** Note that we are not discussing the other important components in the WebSphere Portal infrastructure, such as the network, systems, databases, and so on. We only discuss the WebSphere Portal related metrics that need to be monitored.

5.5.1 What to monitor

It is very important to first understand what exactly needs to be monitored in the WebSphere Portal infrastructure. For a highly available (24X7) environment, we recommend monitoring the following:

- JVM memory usage
- Server Response time
- CPU utilization
- Metrics of all Web applications
- User sessions and details
- Dynacache
- Enterprise JavaBeans (EJBs)
- Thread pools
- Java Database Connectivity (JDBC) Pools
- Custom Application MBeans (JMX™) attributes

5.5.2 Useful resources

More information about monitoring methodologies can be found at:

http://www.ibm.com/developerworks/websphere/library/techarticles/0608_hesmer/0608_hesmer.html

A

Using IBM tools to find solutions and promote customer self-help

The information in this appendix is intended to be a guide on what tools to install and how to use those tools to best enable and promote customer self-help within your organization. Promoting a comprehensive approach on self-help within your organization will greatly improve the efficiency, product skill, and the confidence of your WebSphere Portal Server administration staff.

All of our customers still have access to the IBM traditional world class remote Level 2 defect support teams when required. However, promoting this self-help strategy will enable a staff to be able to effectively and confidently address problems and new requirements in the WebSphere Portal Server environment with increased self-sufficiency. Any improvement in self-sufficiency will greatly increase the chances of reaching your companies' project deadlines on a more consistent basis.

Here we outline the best practices to enable and promote customer self-help:

1. Install IBM Support Assistant.
2. Fundamentally understand the power of the WebSphere Portal Server Support site.
3. Participate actively in the WebSphere Portal Server user comminutes.
4. Register for the WebSphere Portal Server RSS feed.
5. Install the IBM Support Toolbar.
6. Utilize the IBM Education Assistant.
7. Utilize the IBM Guided Activity Assistant.

More details about each of these tools will be provided throughout the remainder of this appendix

> **Important:** WebSphere Portal Server is built on top of WebSphere Application Server, so knowledge of WebSphere Application Server troubleshooting is critical as well. As we describe each of the tools and their best practices, remember to take advantage of the WebSphere Application Server options that are available with each tool.

IBM Support Assistant (ISA)

IBM Support Assistant (ISA) is free and is the IBM premier self-help tool. ISA represents IBM's strategic direction and continued commitment to improving self-help.

ISA is essential to truly enable and promote customer self-help within the organization. Installing ISA alone provides the following components of the best practices to enable and promote customer self-help strategy by enabling quick access to:

- WebSphere Portal Support site
- Newsgroups and forums
- IBM Education Assistant
- IBM Guided Activity Assistant

For complete information about IBM Support Assistant, refer to the ISA Support page at:

http://www-306.ibm.com/software/support/isa/

How does ISA help

The WebSphere Portal Server product team strongly encourages that every customer install and become familiar with this tool. This tool is intended to be the customer's primary entry point into all things that have to do with software support. Using this tool is an essential element for WebSphere Portal Server administrators to be able to stay current with Portal TechNotes, best practices, and any new tools that become available. Also, using ISA will make things much more efficient when a problem occurs and research is required. The tool is especially helpful when a problem requires interaction with the WebSphere Portal Server Level 2 Support team and a PMR and log collection is required.

The way WebSphere Portal Level 2 support does its work is designed around the expectation that customer's understand the functions and benefits of ISA and how to use it in the context of problem determination and the formal support process of opening PMRs. ISA is extremely beneficial in many areas.

Have you ever been frustrated by an error message and unable to quickly find the right solution? ISA provides access to the information you need quickly. ISA provides this quick access through its concurrent Search tool that spans the bulk of IBM documentation and returns the results categorized by source for easy review.

Perhaps you want to utilize a feature within an IBM product, but do not know where to find the relevant how-to documentation regarding this feature. In addition to the Search feature, ISA provides a Product Information feature that links to product education content by leveraging the IBM Education Assistant tool. Other links that are essential to self-help include:

- Product support pages
- Product education roadmaps through the IBM Education Assistant
- Product home pages
- Product recommended updates
- Product troubleshooting and “step-by-step” guides
- Product newsgroups and forums

Do you wish for advanced, easy-to-use tools designed to diagnose errors? Included in ISA is a new Tool workbench that provides you with the problem determination tools that IBM Support itself uses to resolve issues. Going forward, the various product teams will make more and more tools available for ISA's tools framework in order to enable you to perform problem determination on your desktop just like the support team.

ISA also provides a Service feature with an automated system and symptom based collector. The system collector gathers general information and logs from your operating system, registry, and so on. The symptom based collection provides the unique ability to collect specific information and logs relating to a particular problem that you are having. It can also provide you with the ability to automatically enable tracing that will be helpful to IBM support as part of the data gathering process.

Another aspect of the Service feature is the problem submission tool. This allows you to enter your entitlement information once and have it saved for future sessions. You can then easily create a problem report (PMR) for IBM and attach the collector file at the same time. It is simple to do and yet extremely helpful for expediting a solution from IBM.

So whether you need to find information about a software fix, collect key logs, or want to build your skills on a particular product, the IBM Support Assistant can help you get that done.

For more details about the features of ISA, refer to the developerWorks® article, “The Support Authority: Getting help from the IBM Support Assistant”. This article does a exceptional job of explaining the features in detail and how they can be used to enable self-help. It can be found at:

http://www.ibm.com/developerworks/websphere/techjournal/0706_supauth/0706_supauth.html

How do I obtain, install, and access ISA

The ISA tool can be downloaded from the ISA Support page at:

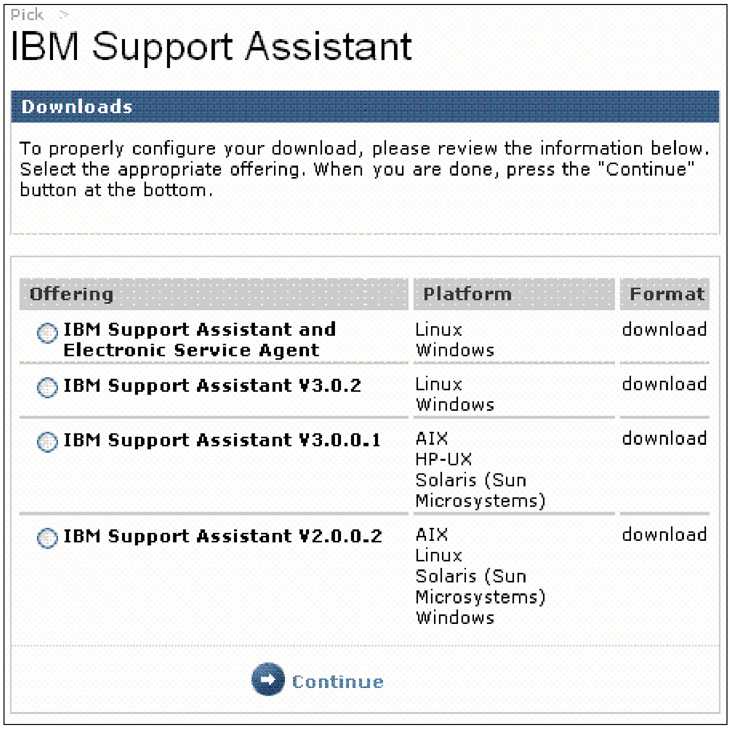
http://www-306.ibm.com/software/support/isa/

Once on the main ISA Support page, you can download the tool by clicking the download link, which takes you to this page:

https://www14.software.ibm.com/webapp/iwm/web/preLogin.do?source=isa

At this page, you be required to either sign in using an existing account or register for a new account. Once signed in, the download page should look something similar to Figure A-1 on page 172.

Note: As of this writing, the IBM Support Assistant and Electronic Service Agent™ represents the newest ISA code, Version 3.1. A request has been submitted to more clearly identify the code level on the page.


Pick >

IBM Support Assistant

Downloads

To properly configure your download, please review the information below. Select the appropriate offering. When you are done, press the "Continue" button at the bottom.

Offering	Platform	Format
○ **IBM Support Assistant and Electronic Service Agent**	Linux Windows	download
○ **IBM Support Assistant V3.0.2**	Linux Windows	download
○ **IBM Support Assistant V3.0.0.1**	AIX HP-UX Solaris (Sun Microsystems)	download
○ **IBM Support Assistant V2.0.0.2**	AIX Linux Solaris (Sun Microsystems) Windows	download

Figure A-1 ISA Download page

Security Considerations: As with any software application, ISA's security greatly depends on the overall security architecture in place on the machine itself. ISA runs as a Web application on a small application server. At startup, the default behavior for the application server is to dynamically pick an open port. The port will usually be different each time ISA starts, but it is possible for a user to configure the application server to use a static port for increased security and control.

As for access, the default configuration for ISA V3 is to only allow access from localhost. Therefore, if the machine itself is secure from the outside, then ISA does nothing to undermine that security.

Before choosing the install code and beginning the download, there a couple thing to consider.

First, you have two options on how to use ISA in your WebSphere Portal Server environment:

- Local: Install ISA on the WebSphere Portal Server machine.
- Remote: Install ISA on a designated ISA administration machine. Note that this is not suggesting that this machine *only* be used for ISA; rather, its simply an administration machine that is remote from the WebSphere Portal Server environment.

Note: The remote option may be desired when the WebSphere Portal Server is running in production and you do not want to place any further resource requirements on the machine that may impact performance, or because of business rules, you are not allowed to install any additional software onto the machine.

Second, once you have decided where ISA will be installed and running, then you have another decision to make based on the OS of the machine that will be running ISA:

- For Windows and Linux

  Download and install ISA V3.1. The ISA V3.1 installer only supports Windows and Linux.

- For HP/UX, Solaris, and AIX

  Download and install ISA V3.0.0.1. Once ISA V3.0.0.1 is installed, then use the Updater to upgrade to ISA 3.0.2

  > **Note:** You may notice that the download page lists V3.0.2 as only for Windows and Linux, but this is only for the installer. You cannot directly install V3.0.2 to AIX, HP/UX, or Solaris. However, once you have V3.0.0.1 installed on AIX, HP/UX, or Solaris, you can use the Updater to upgrade to V3.0.2 because the upgrade path/code supports AIX, HP/UX, and Solaris.
  >
  > As you can see, due to this architecture, systems running on AIX, HP/UX, and Solaris can only be upgraded to the V3.0.2 level, while Windows and Linux users can install V3.1. However, ISA V3.1 does not contain significantly more functionality than V3.0.2, and all fixes found in V3.1 also get back ported to V3.0.2. The same is true for plug-ins. So bottom line is that although AIX, HP/UX, and Solaris user are running a lower level of ISA, the core functionality is the same.

Once the desired ISA install package is downloaded, you will need to extract the install package. Once the package is extracted, you can launch the ISA installer by running the setup script. For details about the ISA install process, refer to the *Installation and Troubleshooting Guide* that is contained in the extracted package along with the install code.

Once ISA is installed, launch ISA.

> **Note:** It is important to understand that the ISA code you just installed is *only* the ISA framework. The full power and features that have been discussed before are contained in the individual product plug-ins. Specific product plug-ins can be added and removed as desired through the Updater feature of the ISA core framework.

After ISA is launched, the first task should be to use the ISA Updater tab to obtain the desired product plug-ins. Once the desired plug-ins are added, ISA is ready to be used to provide self-help.

Again, for more details about performing the install or upgrade, or details about using ISA’s Updater features, refer to the following links:

- “The Support Authority: Getting help from the IBM Support Assistant”, found at:

  `http://www.ibm.com/developerworks/websphere/techjournal/0706_supauth/0706_supauth.html`

- The ISA training from the IBM Education Assistant, found at:

  `http://publib.boulder.ibm.com/infocenter/ieduasst/v1r1m0/topic/com.ibm.iea.isa/isa/isa2_coverpage.html`

- The *Installation and Troubleshooting Guide* that was downloaded with the install package.

Best practices

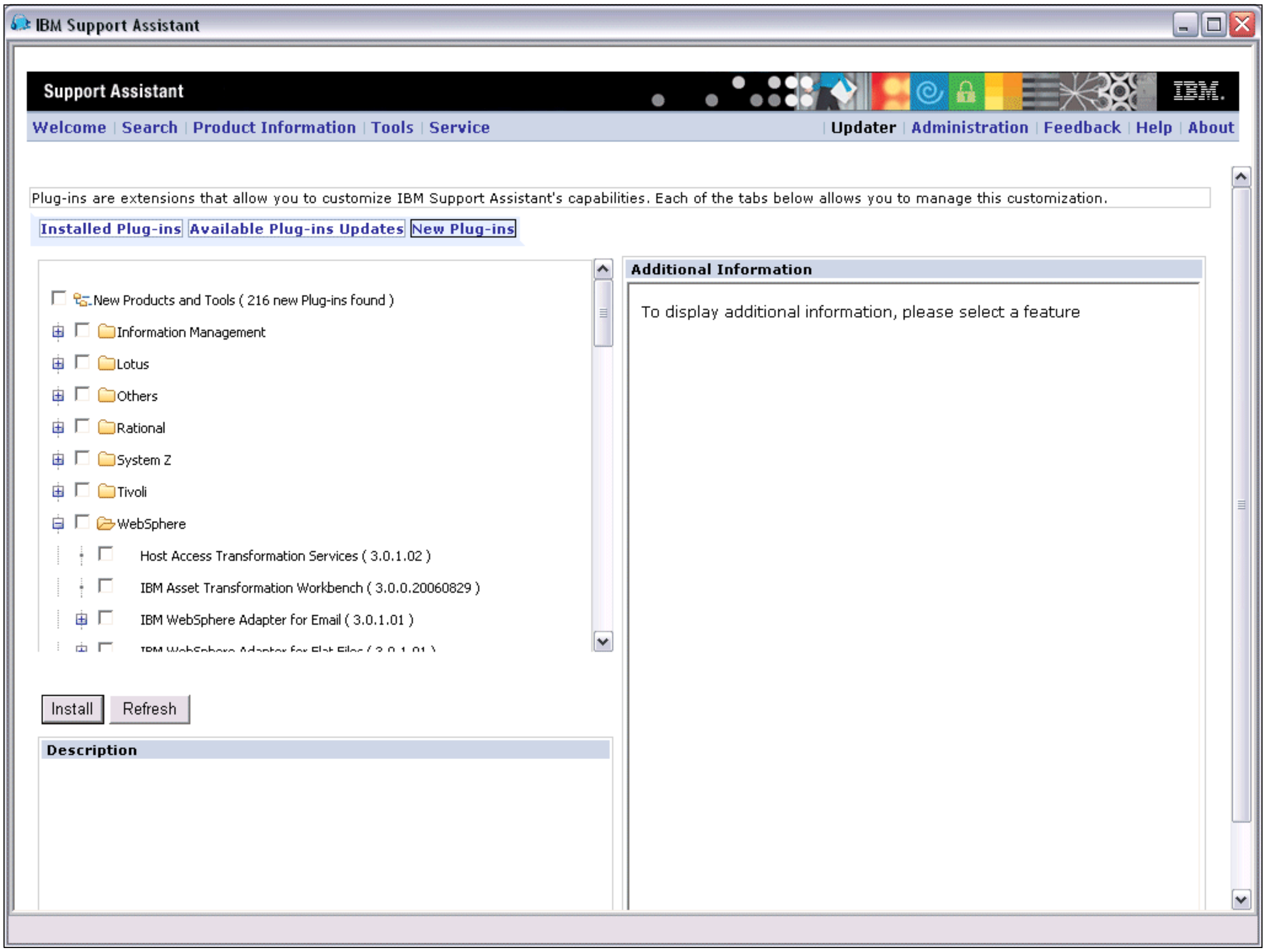
Once ISA has been installed into your environment, the next step is to obtain the desired product plug-ins so you can begin using the full power of the tool to perform research and investigate problems.

As previously mentioned, use the Updater feature to install and update the product plug-ins. Understanding the Updater feature is very important. After ISA is started, click the **Updater** feature to access the available plug-ins. For WebSphere Portal Server V6.0, we recommend that the following plug-ins be installed:

- WebSphere Portal V5.1
- WebSphere Portal V6.0
- WebSphere Application Server V5.1
- WebSphere Application Server V6.0

> **Important:** WebSphere Portal Server is built on top of WebSphere Application Server, so a knowledge of WebSphere Application Server troubleshooting is critical as well. Therefore, we also highly recommend installing the WebSphere Application Server plug-ins as well.

After clicking the **Updater** tab, click the **New Plug-ins** link and expand the WebSphere folder, as shown in Figure A-2.


IBM Support Assistant
Support Assistant
IBM.
Welcome | Search | Product Information | Tools | Service
Updater | Administration | Feedback | Help | About
Installed Plug-ins | Available Plug-ins Updates | New Plug-ins
Additional Information
Install
Refresh
Description

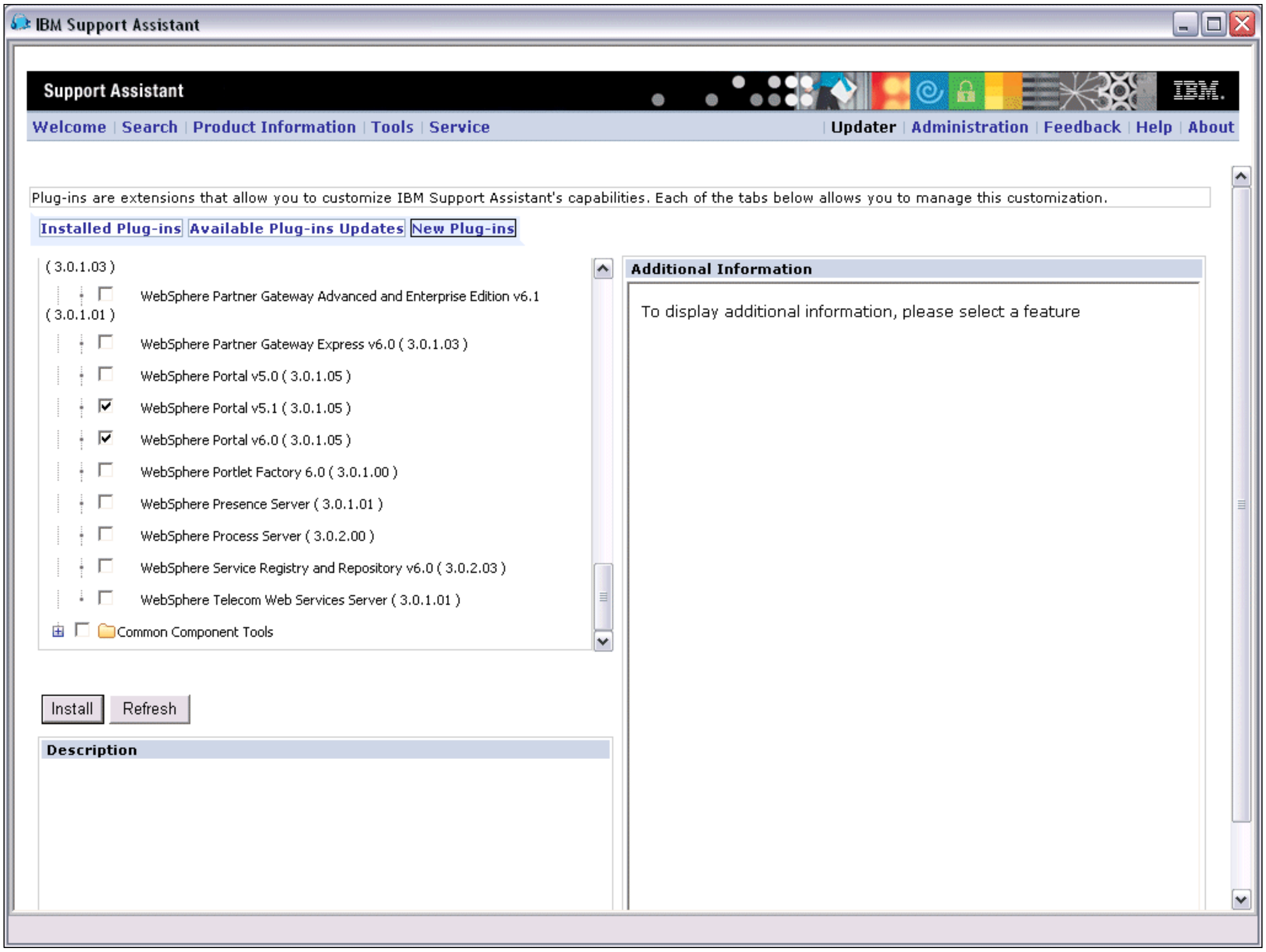
Plug-ins are extensions that allow you to customize IBM Support Assistant's capabilities. Each of the tabs below allows you to manage this customization.

New Products and Tools (216 new Plug-ins found)
Information Management
Lotus
Others
Rational
System Z
Tivoli
WebSphere
Host Access Transformation Services (3.0.1.02)
IBM Asset Transformation Workbench (3.0.0.20060829)
IBM WebSphere Adapter for Email (3.0.1.01)
IBM WebSphere Adapter for Flat Files (3.0.1.01)

To display additional information, please select a feature

Figure A-2 Plug-in list

Next, scroll down and choose the plug-ins listed in Figure A-2 on page 174 and click the **Install** button to install the WebSphere Application Server and WebSphere Portal Server plug-ins. See Figure A-3.


IBM Support Assistant
Support Assistant
IBM.
Welcome | Search | Product Information | Tools | Service
Updater | Administration | Feedback | Help | About
Installed Plug-ins | Available Plug-ins Updates | New Plug-ins
Additional Information
Install
Refresh
Description

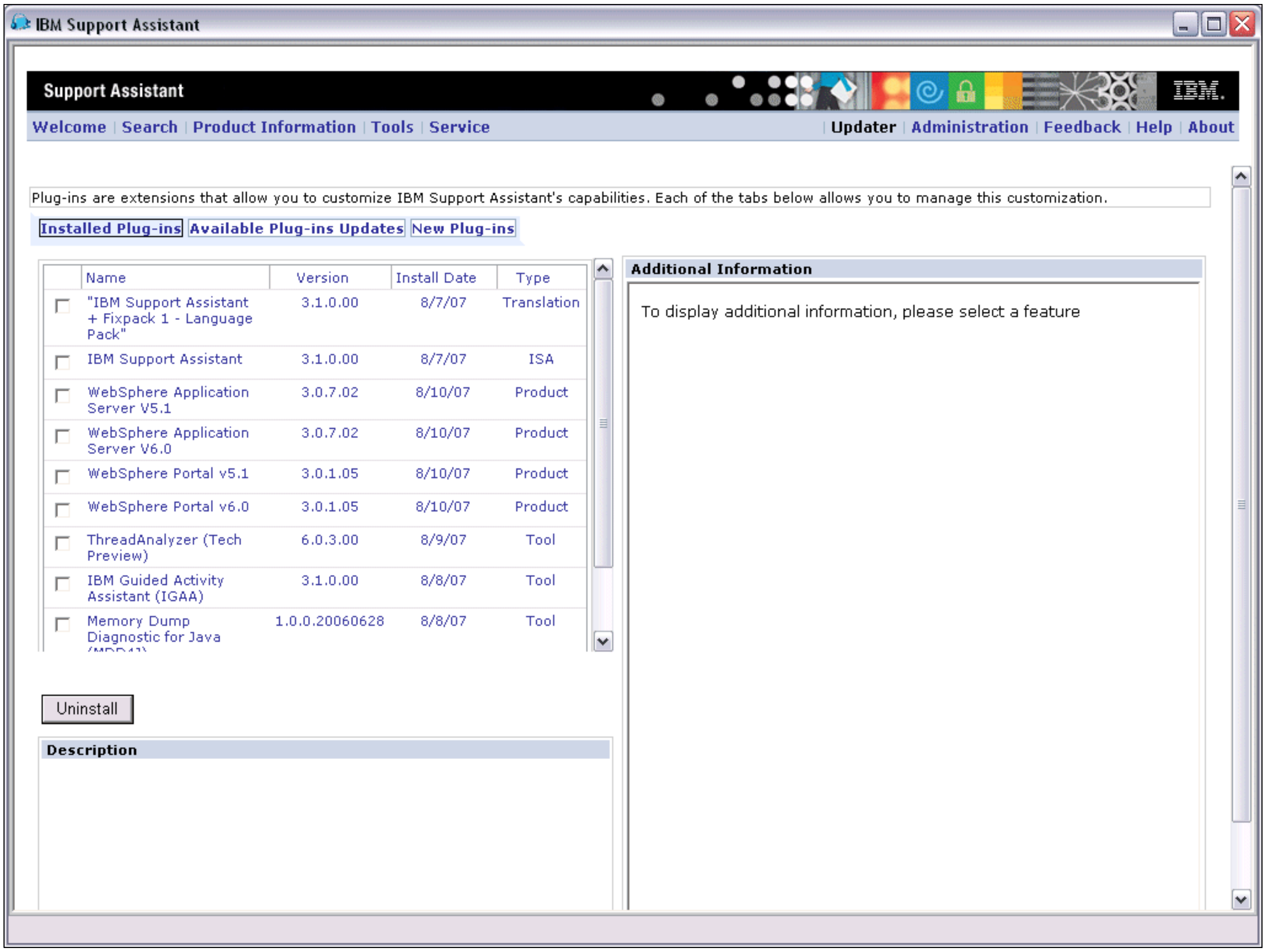
Plug-ins are extensions that allow you to customize IBM Support Assistant's capabilities. Each of the tabs below allows you to manage this customization.

(3.0.1.03)
□ WebSphere Partner Gateway Advanced and Enterprise Edition v6.1 (3.0.1.01)
□ WebSphere Partner Gateway Express v6.0 (3.0.1.03)
□ WebSphere Portal v5.0 (3.0.1.05)
☑ WebSphere Portal v5.1 (3.0.1.05)
☑ WebSphere Portal v6.0 (3.0.1.05)
□ WebSphere Portlet Factory 6.0 (3.0.1.00)
□ WebSphere Presence Server (3.0.1.01)
□ WebSphere Process Server (3.0.2.00)
□ WebSphere Service Registry and Repository v6.0 (3.0.2.03)
□ WebSphere Telecom Web Services Server (3.0.1.01)
□ Common Component Tools

To display additional information, please select a feature

Figure A-3 Install plug-ins

Once the plug-ins are installed, the ISA tool will prompt you for a restart. Once ISA has been restarted, you can navigate to the Updater feature again and click **Installed Plug-ins**, and you should see the four plug-ins we just installed. See Figure A-4.


IBM Support Assistant
Support Assistant
IBM.
Welcome | Search | Product Information | Tools | Service
Updater | Administration | Feedback | Help | About
Installed Plug-ins | Available Plug-ins Updates | New Plug-ins
Uninstall
Description
Additional Information

Plug-ins are extensions that allow you to customize IBM Support Assistant's capabilities. Each of the tabs below allows you to manage this customization.

	Name	Version	Install Date	Type
□	"IBM Support Assistant + Fixpack 1 - Language Pack"	3.1.0.00	8/7/07	Translation
□	IBM Support Assistant	3.1.0.00	8/7/07	ISA
□	WebSphere Application Server V5.1	3.0.7.02	8/10/07	Product
□	WebSphere Application Server V6.0	3.0.7.02	8/10/07	Product
□	WebSphere Portal v5.1	3.0.1.05	8/10/07	Product
□	WebSphere Portal v6.0	3.0.1.05	8/10/07	Product
□	ThreadAnalyzer (Tech Preview)	6.0.3.00	8/9/07	Tool
□	IBM Guided Activity Assistant (IGAA)	3.1.0.00	8/8/07	Tool
□	Memory Dump Diagnostic for Java	1.0.0.20060628	8/8/07	Tool

To display additional information, please select a feature

Figure A-4 List of installed plug-ins

Again, using the Updater feature is very important. Continue to check the Updater regularly for updates to existing product plug-ins and also updates to the ISA core itself. Watch the New Plug-ins link to see if any new plug-ins may be of value to you and your WebSphere Portal Server environment.

The next best practice step is to get into the habit of opening ISA each morning you begin work. Get in the habit of using ISA as your interface/access into the world of WebSphere Portal Server support.

Initially, you will most likely find these features particularly helpful:

- Search: The federated search capability is very powerful and intuitive. Give it a try and see for yourself.
- Product Information: This feature allows quick access to the WebSphere Portal Server forums. Forums are a great place to find answers from the collective WebSphere Portal Server user community.
- Tools: We recommend to immediately use the Updater to install the IBM Guided Activity Assistant (IGAA). Once the IGAA plug-in is installed, you can launch it directly from the ISA Tools feature. IGAA gives you instant access to many helpful guided activities.

Note: At the time of this writing, no IGAA content exists for WebSphere Portal Server. However, there is a very useful WebSphere Application Server Troubleshooting guided activity available that can be obtained and installed through the ISA interface. To obtain a guided activity, simply launch the IGAA from the ISA Tools Feature and then the IGAA will guide you.

- Service: The service feature can be used to create a PMR through ESR, and also provide the ability to automate log collection.
- IBM Workplace for Customer Support: If you are using ISA V3.1 and you are a Premium Support Customer, you can launch the IBM Workplace for Customer Support page by clicking on the icon on ISA's Welcome page. From this page, you can click a link to launch the IBM Workplace for Customer Support portal.

Use case examples - Search

The Search feature of ISA is one of ISA's most powerful features and will most likely be the feature that is initially used the most for self-help. The Search feature can be used in two ways: problem avoidance and problem determination.

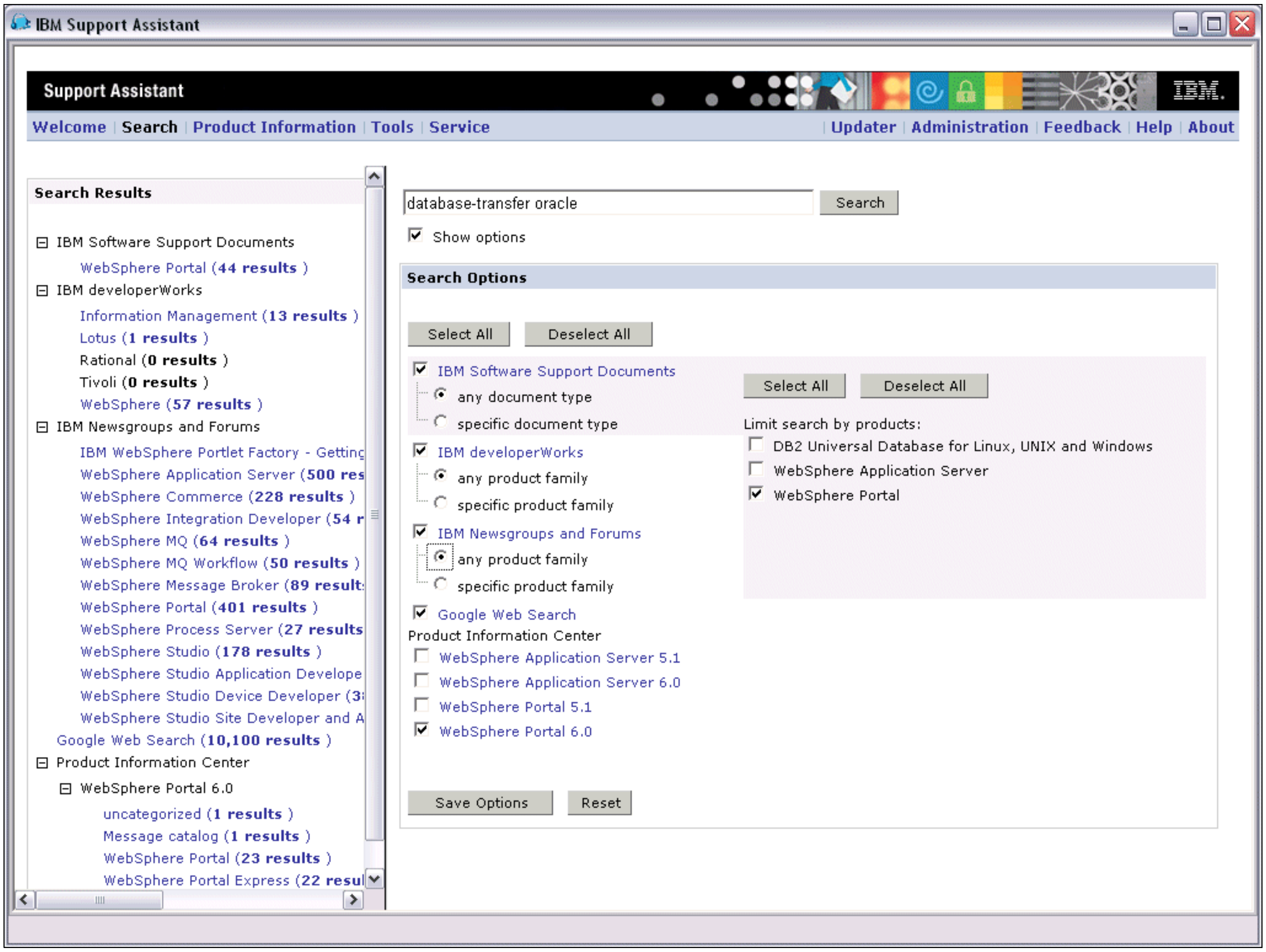
A typical problem avoidance scenario may be something like the following. You have installed WebSphere Portal Server and your next task is to transfer the database from CloudScape to Oracle. Before starting this procedure, you want to try to understand any known pitfalls that may cause problems so you can avoid them if possible.

So in this example, we will use the Search feature to search for the string, "database-transfer oracle" by entering the error code in the search box and then selecting the scope of the search. For this example, we have chosen to search all repositories:

- IBM Software Support Documents
- IBM Developer Works
- IBM Newsgroups and Forums
- Google
- Product Information Centers

Since the error is occurring in WebSphere Portal Server V6.0.x, we have limited the IBM Software Support Documents search to only WebSphere Portal Server. In addition, we are only searching the WebSphere Portal Server V6.0 InfoCenter. This showcases the power of ISA's Search feature, as you can narrow or broaden the scope of your searches in many different ways based on need and environment.

From the results listed in Figure A-5, you can investigate further for any known problems to avoid.



IBM Support Assistant
Support Assistant
IBM.
Welcome | Search | Product Information | Tools | Service | Updater | Administration | Feedback | Help | About

Search Results

- ⊟ IBM Software Support Documents
  - WebSphere Portal (**44 results**)
- ⊟ IBM developerWorks
  - Information Management (**13 results**)
  - Lotus (**1 results**)
  - Rational (**0 results**)
  - Tivoli (**0 results**)
  - WebSphere (**57 results**)
- ⊟ IBM Newsgroups and Forums
  - IBM WebSphere Portlet Factory - Getting
  - WebSphere Application Server (**500 res**
  - WebSphere Commerce (**228 results**)
  - WebSphere Integration Developer (**54 r**
  - WebSphere MQ (**64 results**)
  - WebSphere MQ Workflow (**50 results**)
  - WebSphere Message Broker (**89 result**
  - WebSphere Portal (**401 results**)
  - WebSphere Process Server (**27 results**
  - WebSphere Studio (**178 results**)
  - WebSphere Studio Application Develope
  - WebSphere Studio Device Developer (3
  - WebSphere Studio Site Developer and A
- Google Web Search (**10,100 results**)
- ⊟ Product Information Center
  - ⊟ WebSphere Portal 6.0
    - uncategorized (**1 results**)
    - Message catalog (**1 results**)
    - WebSphere Portal (**23 results**)
    - WebSphere Portal Express (**22 resul**

database-transfer oracle [Search]

☑ Show options

Search Options

[Select All] [Deselect All]

☑ IBM Software Support Documents
- ● any document type
- ○ specific document type

[Select All] [Deselect All]

Limit search by products:
- ☐ DB2 Universal Database for Linux, UNIX and Windows
- ☐ WebSphere Application Server
- ☑ WebSphere Portal

☑ IBM developerWorks
- ● any product family
- ○ specific product family

☑ IBM Newsgroups and Forums
- ● any product family
- ○ specific product family

☑ Google Web Search

Product Information Center
- ☐ WebSphere Application Server 5.1
- ☐ WebSphere Application Server 6.0
- ☐ WebSphere Portal 5.1
- ☑ WebSphere Portal 6.0

[Save Options] [Reset]

Figure A-5 Search for known problems

A typical problem determination scenario may be something like the following. The WebSphere Portal Server fails to start correctly after a successful database-transfer. Using the troubleshooting techniques described in previous chapters, you have determined what you believe to be the most relevant and significant error stack from the logs. The error stack you are focused on is shown here:

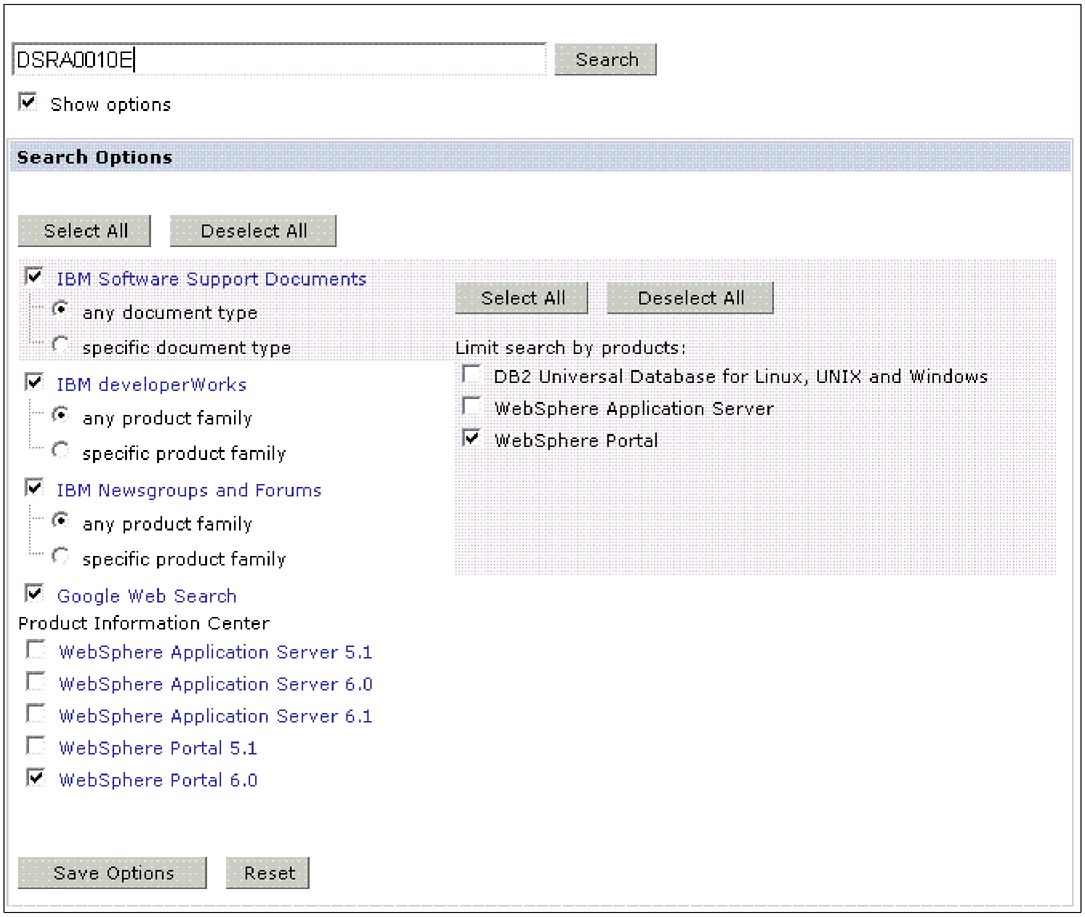
```
Caused by: java.sql.SQLException: Database 'wp601' not found.DSRA0010E: SQL State
= XJ004, Error Code = 40,000DSRA0010E: SQL State = XJ004, Error Code = 40,000
   at db2j.ai.j._f20(Unknown Source)
   at db2j.ai.j.newCloudscapeSQLException(Unknown Source)
   at db2j.ai.j.generateCsSQLException(Unknown Source)
   at db2j.ai.c.<init>(Unknown Source)
   at db2j.ax.b.<init>(Unknown Source)
   at db2j.aw.c.<init>(Unknown Source)
```

Using this information, let us use the ISA Search feature to see what we can find.

In this example, we will search for the error code DSRA0010E by entering the error code in the Search box and then selecting the scope of the search. Again, we have chosen to search all repositories:

- IBM Software Support Documents
- IBM Developer Works
- IBM Newsgroups and Forums
- Google
- Product Information Centers

Since the error is occurring on WebSphere Portal Server V6.0.x, we have limited the IBM Software Support Documents search to only WebSphere Portal Server. In addition, we are only searching the WebSphere Portal Server V6.0 InfoCenter, as shown in Figure A-6. Once again, this showcases the power of ISA's Search feature, as you can narrow or broaden the scope of your searches in many different ways based on need and environment.



DSRA0010E

☑ Show options

☑ IBM Software Support Documents
● any document type
○ specific document type

Limit search by products:
□ DB2 Universal Database for Linux, UNIX and Windows
□ WebSphere Application Server
☑ WebSphere Portal

☑ IBM developerWorks
● any product family
○ specific product family

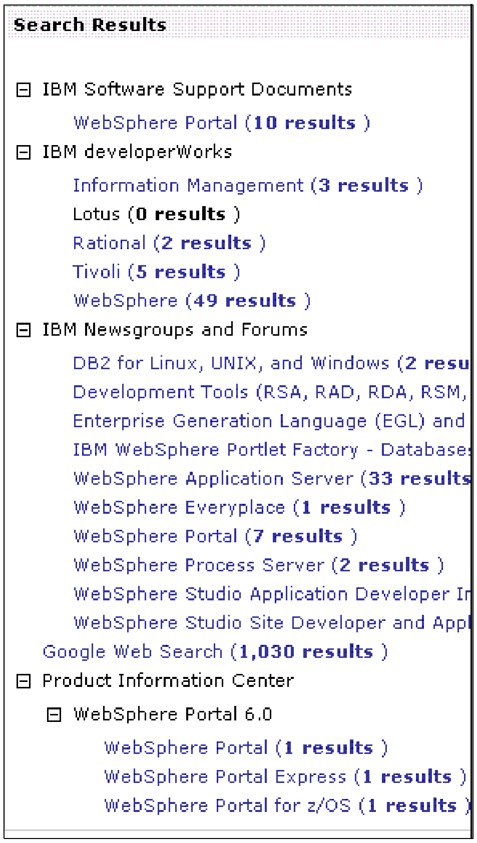
☑ IBM Newsgroups and Forums
● any product family
○ specific product family

☑ Google Web Search

Product Information Center
□ WebSphere Application Server 5.1
□ WebSphere Application Server 6.0
□ WebSphere Application Server 6.1
□ WebSphere Portal 5.1
☑ WebSphere Portal 6.0

Figure A-6 Initial search string

After selecting these search options, click **Search** and wait for ISA to populate the results in the left hand pane, as shown in Figure A-7.



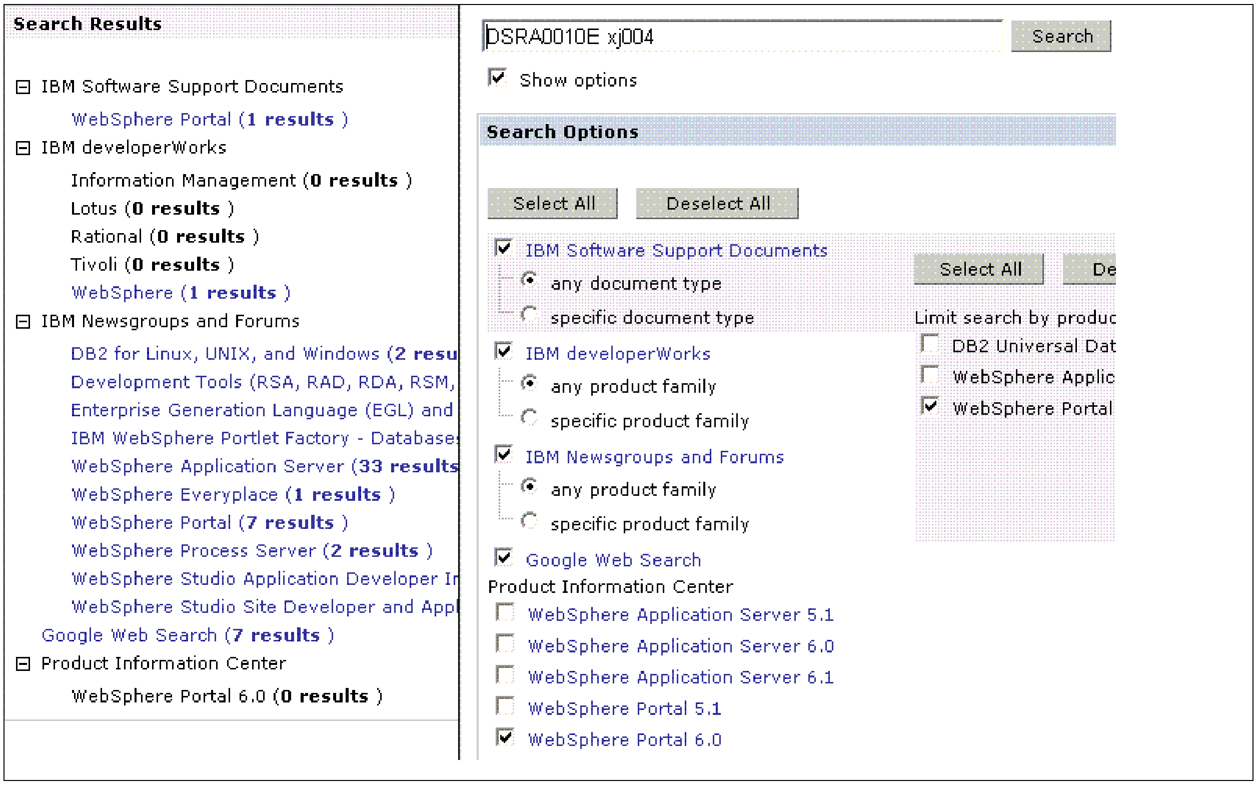
Search Results

- ⊟ IBM Software Support Documents
  - WebSphere Portal (**10 results**)
- ⊟ IBM developerWorks
  - Information Management (**3 results**)
  - Lotus (**0 results**)
  - Rational (**2 results**)
  - Tivoli (**5 results**)
  - WebSphere (**49 results**)
- ⊟ IBM Newsgroups and Forums
  - DB2 for Linux, UNIX, and Windows (**2 resu**
  - Development Tools (RSA, RAD, RDA, RSM,
  - Enterprise Generation Language (EGL) and
  - IBM WebSphere Portlet Factory - Database
  - WebSphere Application Server (**33 results**
  - WebSphere Everyplace (**1 results**)
  - WebSphere Portal (**7 results**)
  - WebSphere Process Server (**2 results**)
  - WebSphere Studio Application Developer In
  - WebSphere Studio Site Developer and Appl
- Google Web Search (**1,030 results**)
- ⊟ Product Information Center
  - ⊟ WebSphere Portal 6.0
    - WebSphere Portal (**1 results**)
    - WebSphere Portal Express (**1 results**)
    - WebSphere Portal for z/OS (**1 results**)

Figure A-7 Initial search results

As you can see in Figure A-7, the search returns items from each repository and lists the search results out separately.

So at this point, let us see if we can narrow the results. So, we use the same settings, but we further qualify the search with the string “DSRA0010E xj004”, as shown in Figure A-8.



Search Results

⊟ IBM Software Support Documents
- WebSphere Portal (**1 results**)

⊟ IBM developerWorks
- Information Management (**0 results**)
- Lotus (**0 results**)
- Rational (**0 results**)
- Tivoli (**0 results**)
- WebSphere (**1 results**)

⊟ IBM Newsgroups and Forums
- DB2 for Linux, UNIX, and Windows (**2 resu**
- Development Tools (RSA, RAD, RDA, RSM,
- Enterprise Generation Language (EGL) and
- IBM WebSphere Portlet Factory - Database
- WebSphere Application Server (**33 results**
- WebSphere Everyplace (**1 results**)
- WebSphere Portal (**7 results**)
- WebSphere Process Server (**2 results**)
- WebSphere Studio Application Developer In
- WebSphere Studio Site Developer and Appl

Google Web Search (**7 results**)

⊟ Product Information Center
- WebSphere Portal 6.0 (**0 results**)

DSRA0010E xj004

☑ Show options

Search Options

☑ IBM Software Support Documents
- ● any document type
- ○ specific document type

Limit search by produc
- ☐ DB2 Universal Dat
- ☐ WebSphere Applic
- ☑ WebSphere Portal

☑ IBM developerWorks
- ● any product family
- ○ specific product family

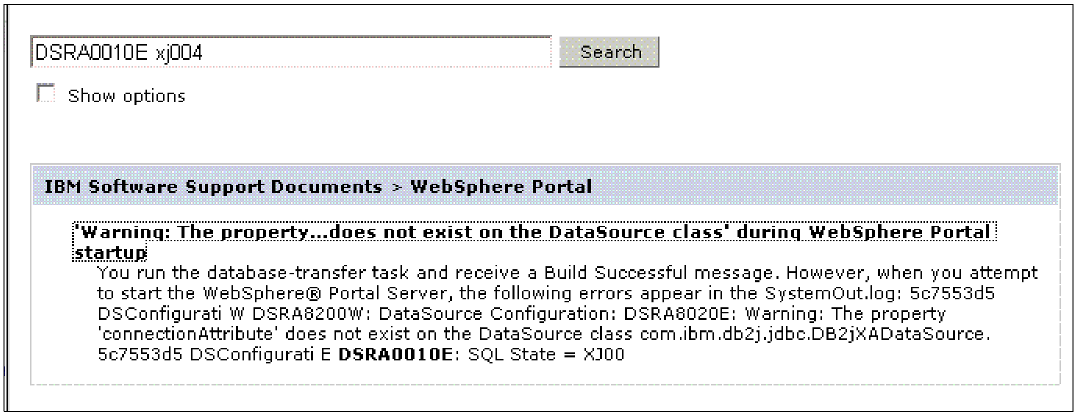
☑ IBM Newsgroups and Forums
- ● any product family
- ○ specific product family

☑ Google Web Search

Product Information Center
- ☐ WebSphere Application Server 5.1
- ☐ WebSphere Application Server 6.0
- ☐ WebSphere Application Server 6.1
- ☐ WebSphere Portal 5.1
- ☑ WebSphere Portal 6.0

Figure A-8 Narrowed search string and results

As you see in Figure A-8, we now have one result under IBM Software Support Documents. Let us check that result first since it is searching TechNotes. So we click the result under IBM Software Support Documents and it shows the search results in the right hand pane, as shown in Figure A-9.



DSRA0010E xj004

☐ Show options

IBM Software Support Documents > WebSphere Portal

'Warning: The property...does not exist on the DataSource class' during WebSphere Portal startup

You run the database-transfer task and receive a Build Successful message. However, when you attempt to start the WebSphere® Portal Server, the following errors appear in the SystemOut.log: 5c7553d5 DSConfigurati W DSRA8200W: DataSource Configuration: DSRA8020E: Warning: The property 'connectionAttribute' does not exist on the DataSource class com.ibm.db2j.jdbc.DB2jXADataSource. 5c7553d5 DSConfigurati E **DSRA0010E**: SQL State = XJ00

Figure A-9 TechNote search results

Clicking the TechNote link will open up a Web browser and take you to the following TechNote:

`http://www-1.ibm.com/support/docview.wss?uid=swg21235608`

As you can see, the TechNote seems to describe the same problem and seems to be a good candidate for the resolutions of the problem.

For further details about the individual features offered by ISA, refer to the document, “The Support Authority: Getting help from the IBM Support Assistant”. The particularly useful guide can be found at:

http://www.ibm.com/developerworks/websphere/techjournal/0706_supauth/0706_supauth.html

Use case examples - Product Information

The Product Information feature is intended to be used as an area that the WebSphere Portal Server administrator can go to quickly link to other valuable WebSphere Portal Server sites, step-by-step guides, or other WebSphere Portal Server communities. The most powerful component of this feature is the direct access to the WebSphere Portal Server newsgroups and forums.

Using the WebSphere Portal Server forum is a powerful collaboration mechanism. By accessing the forum, you now have access to the knowledge and experience of the collective WebSphere Portal Server user community. Once in the WebSphere Portal Server forum, you can post a question to the entire WebSphere Portal Server user community for advice or tips and unleash the power of true collaboration.

> **Attention:** The WebSphere Portal Server forum is monitored by members of the WebSphere Portal Server product teams. Active participation by our customers in these user communities is strongly encouraged and recommended.
>
> The newsgorups and forums are a great resource for planning, guidance, tips, troubleshooting techniques, and, most of all, knowledge sharing.

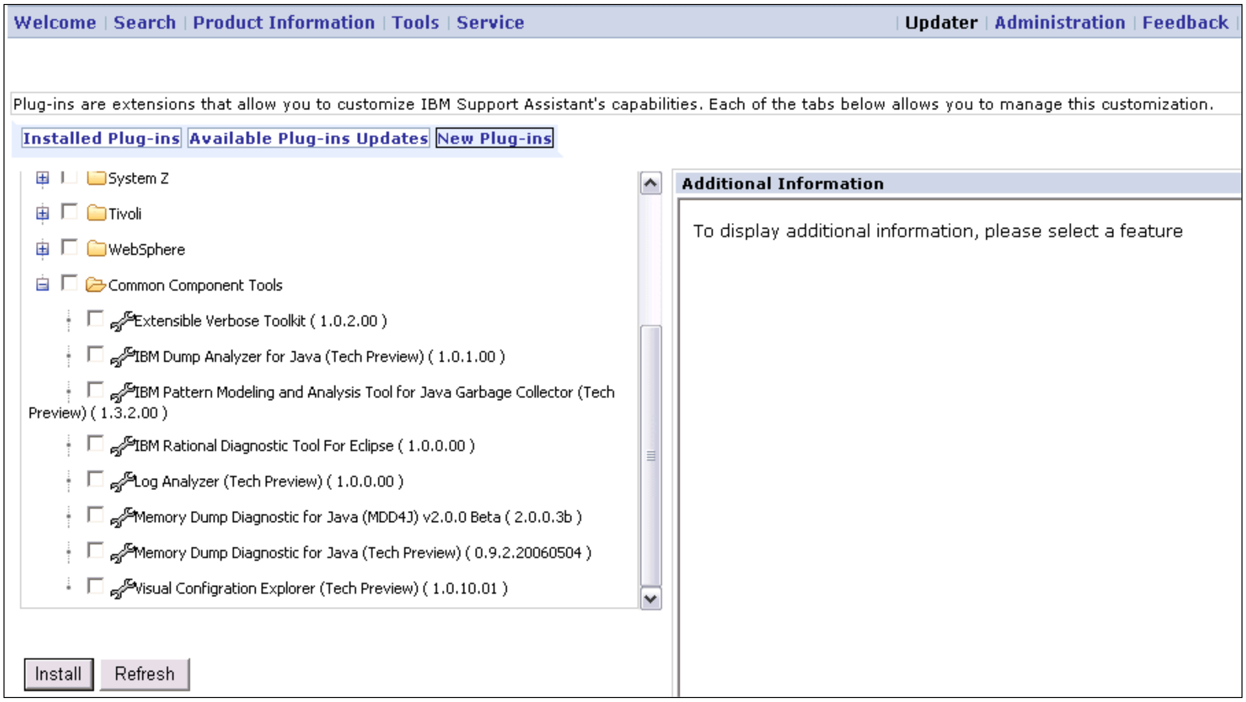
For further details about the individual features offered by ISA, refer the document, “The Support Authority: Getting help from the IBM Support Assistant”. The particularly useful guide can be found at:

http://www.ibm.com/developerworks/websphere/techjournal/0706_supauth/0706_supauth.html

Use case examples - Tools

The Tools feature can be used to access some of the same tooling that IBM Level 2 support uses to troubleshoot problems.

To gain access to the available tools, you must first install the individual tool plug-ins by using the Updater feature. Once in the Updater feature, navigate to the New Plug-ins area and expand the Common Component Tools to see the list of available tool plug-ins. See Figure A-10.


Welcome | Search | Product Information | Tools | Service
Updater | Administration | Feedback
Installed Plug-ins
Available Plug-ins Updates
New Plug-ins
Additional Information
Install
Refresh

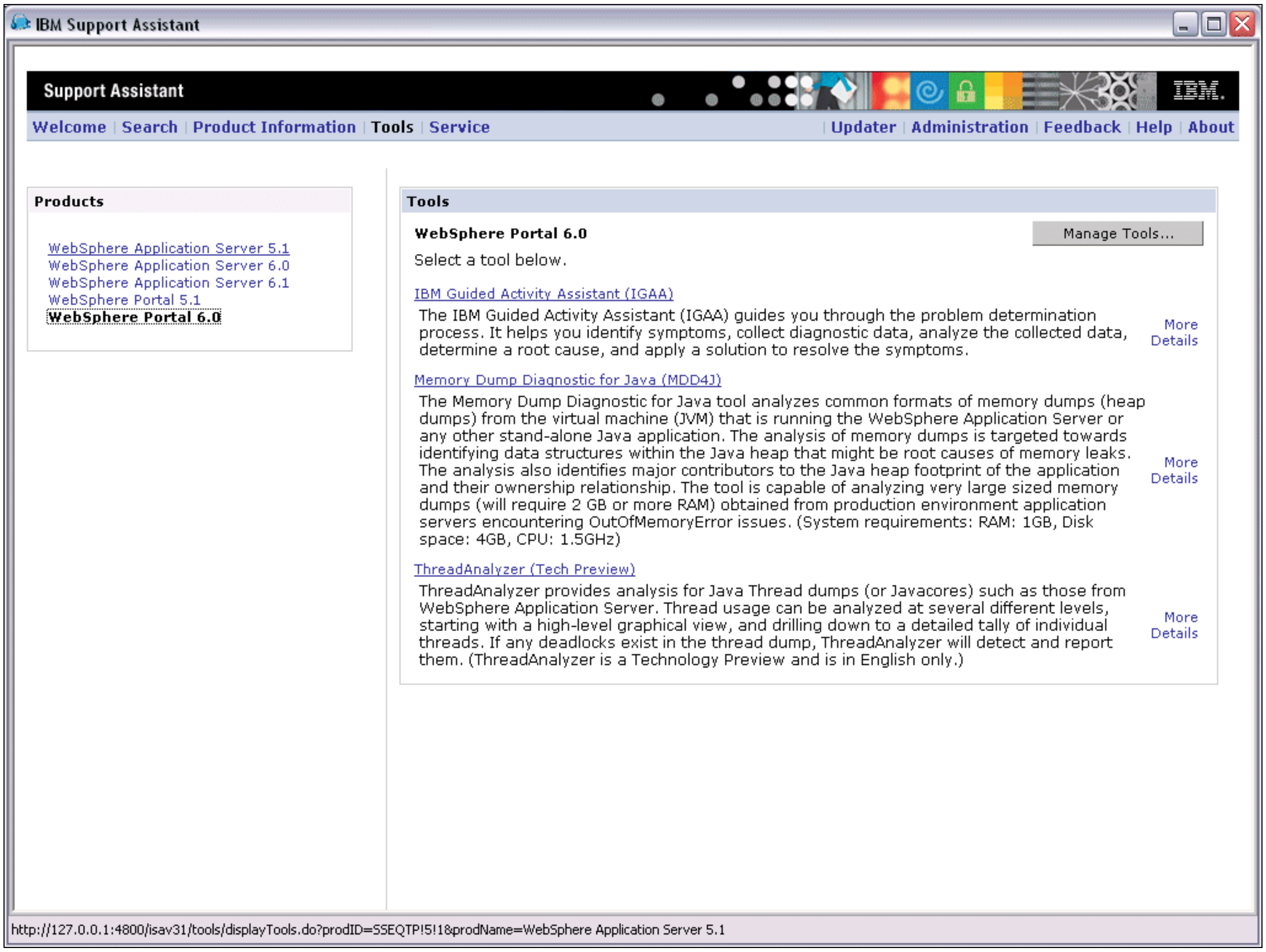
Plug-ins are extensions that allow you to customize IBM Support Assistant's capabilities. Each of the tabs below allows you to manage this customization.

- ⊞ □ System Z
- ⊞ □ Tivoli
- ⊞ □ WebSphere
- ⊟ □ Common Component Tools
  - □ Extensible Verbose Toolkit (1.0.2.00)
  - □ IBM Dump Analyzer for Java (Tech Preview) (1.0.1.00)
  - □ IBM Pattern Modeling and Analysis Tool for Java Garbage Collector (Tech Preview) (1.3.2.00)
  - □ IBM Rational Diagnostic Tool For Eclipse (1.0.0.00)
  - □ Log Analyzer (Tech Preview) (1.0.0.00)
  - □ Memory Dump Diagnostic for Java (MDD4J) v2.0.0 Beta (2.0.0.3b)
  - □ Memory Dump Diagnostic for Java (Tech Preview) (0.9.2.20060504)
  - □ Visual Configration Explorer (Tech Preview) (1.0.10.01)

To display additional information, please select a feature

Figure A-10 Available tool plug-ins

Once the tool plug-in has been installed, the tool will be listed under the Tools feature, as shown in Figure A-11.


IBM Support Assistant
Support Assistant
Welcome | Search | Product Information | Tools | Service
Updater | Administration | Feedback | Help | About
IBM.

Products

- WebSphere Application Server 5.1
- WebSphere Application Server 6.0
- WebSphere Application Server 6.1
- WebSphere Portal 5.1
- **WebSphere Portal 6.0**

Tools

WebSphere Portal 6.0 Manage Tools...

Select a tool below.

IBM Guided Activity Assistant (IGAA)

The IBM Guided Activity Assistant (IGAA) guides you through the problem determination process. It helps you identify symptoms, collect diagnostic data, analyze the collected data, determine a root cause, and apply a solution to resolve the symptoms. More Details

Memory Dump Diagnostic for Java (MDD4J)

The Memory Dump Diagnostic for Java tool analyzes common formats of memory dumps (heap dumps) from the virtual machine (JVM) that is running the WebSphere Application Server or any other stand-alone Java application. The analysis of memory dumps is targeted towards identifying data structures within the Java heap that might be root causes of memory leaks. The analysis also identifies major contributors to the Java heap footprint of the application and their ownership relationship. The tool is capable of analyzing very large sized memory dumps (will require 2 GB or more RAM) obtained from production environment application servers encountering OutOfMemoryError issues. (System requirements: RAM: 1GB, Disk space: 4GB, CPU: 1.5GHz) More Details

ThreadAnalyzer (Tech Preview)

ThreadAnalyzer provides analysis for Java Thread dumps (or Javacores) such as those from WebSphere Application Server. Thread usage can be analyzed at several different levels, starting with a high-level graphical view, and drilling down to a detailed tally of individual threads. If any deadlocks exist in the thread dump, ThreadAnalyzer will detect and report them. (ThreadAnalyzer is a Technology Preview and is in English only.) More Details

http://127.0.0.1:4800/isav31/tools/displayTools.do?prodID=SSEQTP!5!1&prodName=WebSphere Application Server 5.1

Figure A-11 Available tools

For further details about the individual features offered by ISA refer the document, “The Support Authority: Getting help from the IBM Support Assistant”, found at:

http://www.ibm.com/developerworks/websphere/techjournal/0706_supauth/0706_supauth.html

Use case examples - Service

The Service feature provides two very important functions:

- Proactively collects logs using the embedded Automated Problem Determination (AutoPD) log collection mechanism
- Opens new PMRs through the embedded Electronic Service Request (ESR) mechanism

The Service feature should be used once all other efforts to resolve or rediscover the current problem using previously discussed self-help techniques have been exhausted. If these self-help efforts do not identify the solution, then the next step is to formally engage WebSphere Portal Server support through the PMR process.

Once it is determined that a new PMR should be opened, you should first collect the logs. Once the logs are collected, create the PMR and attach the logs to the new PMR.

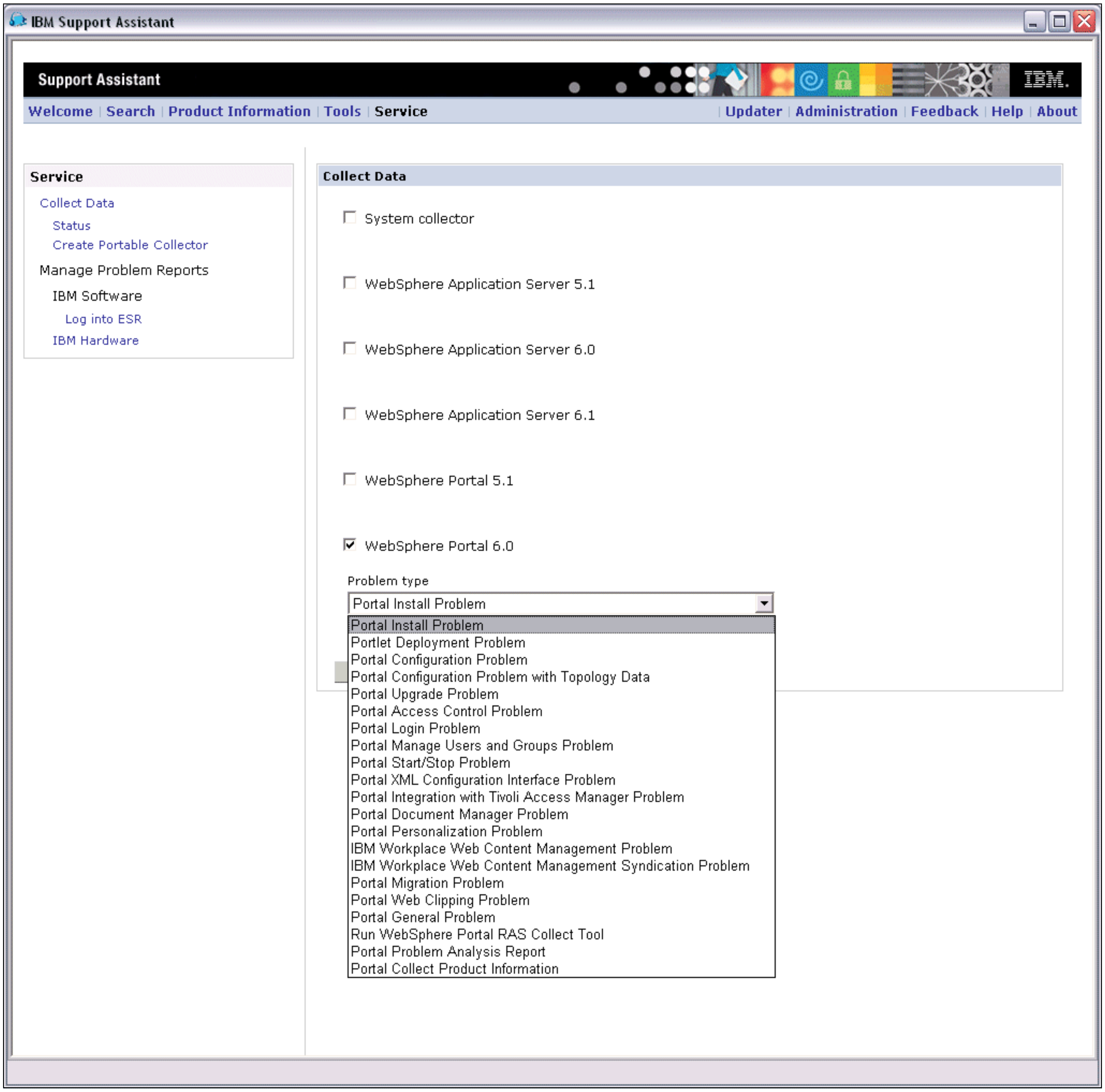
The approach on how to collect logs depends on how you decided to implement ISA into your environment:

- Local: If you installed ISA into the same box as the WebSphere Portal Server, then you can simply use the ISA interface to collect the logs. To collect logs through the ISA interface, the ISA tool must be installed on the same machine as the WebSphere Portal Server.
- Remote: If you installed ISA on an administration machine (a machine that is remote to WebSphere Portal Server), then you must use the ISA Portable Collector.

The typical scenario for collecting logs through a local ISA would look something like this:

- A problem occurs in the WebSphere Portal Server environment.
- You attempt to use self-help techniques and tools to resolve or rediscover the problem and determine a solution.
- If self-help techniques fail to resolve the problem, then the next step is to open a PMR with WebSphere Portal Server support.
- To engage WebSphere Portal Server support, use the Service feature within ISA to first collect the logs using the log collection mechanism. From the drop-down menu, choose the most relevant problem scenario. See Figure A-12 on page 186.

> **Note:** Review the list of collection scripts and choose the one that most fits the nature of your problem. Each collection is custom designed for a specific scenario.
>
> If you are unsure, select **Portal General Problem**. If additional collection scripts are needed, the Level 2 support engineer who takes ownership of your PMR will provide further instructions on the specific script to be run to collect the logs.


IBM Support Assistant
Support Assistant
IBM.
Welcome | Search | Product Information | Tools | Service
| Updater | Administration | Feedback | Help | About
Service
Collect Data
Status
Create Portable Collector
Manage Problem Reports
IBM Software
Log into ESR
IBM Hardware

Collect Data

□ System collector

□ WebSphere Application Server 5.1

□ WebSphere Application Server 6.0

□ WebSphere Application Server 6.1

□ WebSphere Portal 5.1

☑ WebSphere Portal 6.0

Problem type: Portal Install Problem

- Portal Install Problem
- Portlet Deployment Problem
- Portal Configuration Problem
- Portal Configuration Problem with Topology Data
- Portal Upgrade Problem
- Portal Access Control Problem
- Portal Login Problem
- Portal Manage Users and Groups Problem
- Portal Start/Stop Problem
- Portal XML Configuration Interface Problem
- Portal Integration with Tivoli Access Manager Problem
- Portal Document Manager Problem
- Portal Personalization Problem
- IBM Workplace Web Content Management Problem
- IBM Workplace Web Content Management Syndication Problem
- Portal Migration Problem
- Portal Web Clipping Problem
- Portal General Problem
- Run WebSphere Portal RAS Collect Tool
- Portal Problem Analysis Report
- Portal Collect Product Information

Figure A-12 Log collection list

- Once the logs are collected, use the **login to ESR** link to launch the ESR tool to create a new PMR. During this process, you will be allowed to attach the previous log collection to the PMR. By doing this task, the logs will be made available to the support team at the time the PMR is opened.

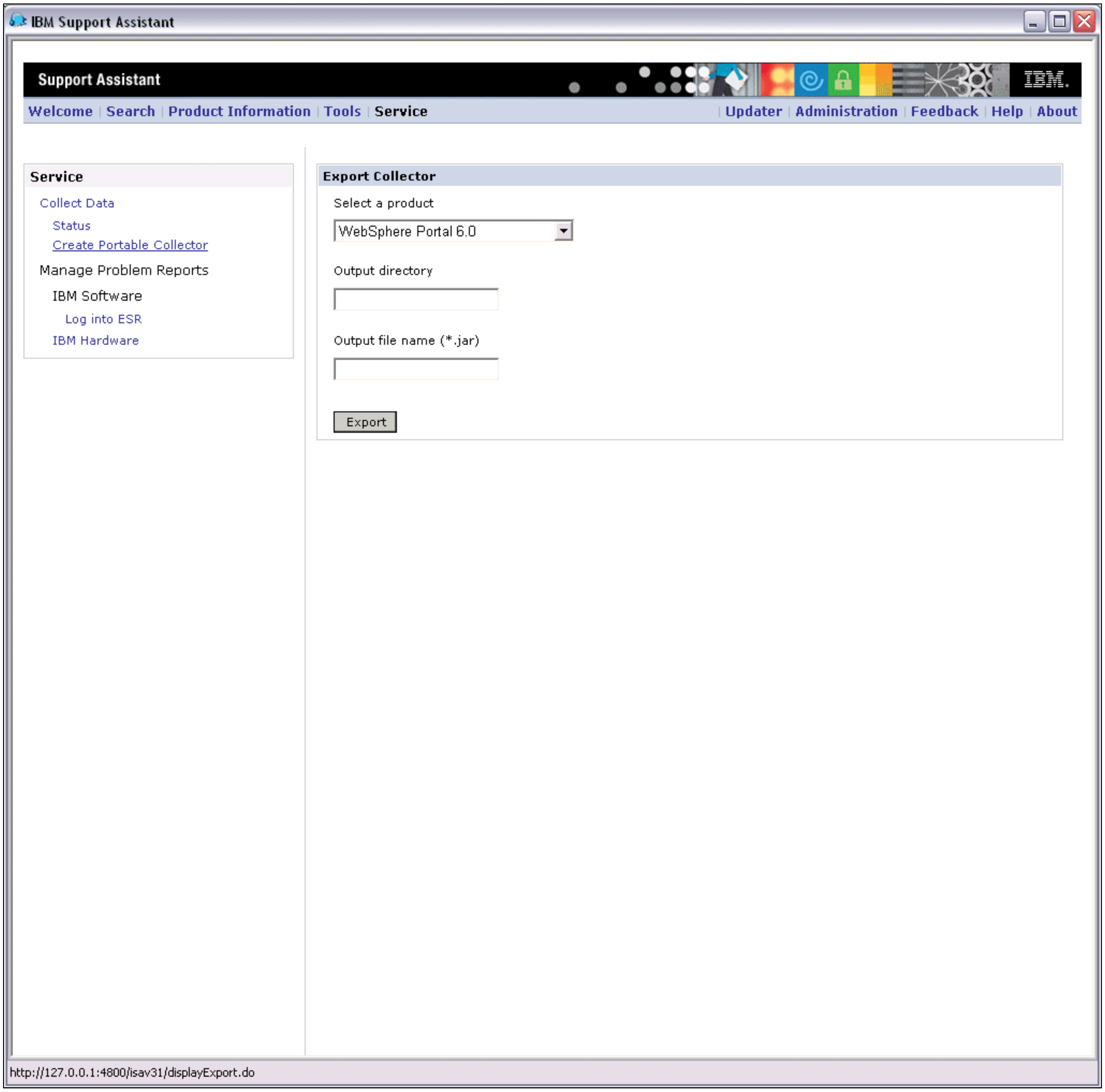
> **Attention:** Following this approach to attach the logs during PMR submission will greatly increase the ability for the Level 2 support engineer to immediately begin work on your PMR. In most cases, following this approach will also result in much quicker problem resolution.

The typical scenario for collecting logs through a remote ISA using the Portable Collector would be similar to the following:

- A problem occurs in the WebSphere Portal Server environment.
- You attempt to use self-help techniques and tools to resolve or rediscover the problem and determine a solution.
- If self-help techniques fail to resolve the problem, then the next step is to open a PMR with WebSphere Portal Server support.
- To engage WebSphere Portal Server support, use the Service feature within ISA to first collect the logs. Since the ISA install is remote to the WebSphere Portal Server installation, you will need to use the Portable Collector.

  Note: A summary of this approach includes using the ISA interface to create the Portable Collector on the remote ISA machine, as shown in Figure A-13. After the <collector>.jar is created on the ISA machine, move it to the remote WebSphere Portal Server machine and run it there.

- Create the Portable Collector by clicking the **Create Portable Collector** option.


IBM Support Assistant
Support Assistant
IBM.
Welcome | Search | Product Information | Tools | Service
Updater | Administration | Feedback | Help | About
Service
Collect Data
Status
Create Portable Collector
Manage Problem Reports
IBM Software
Log into ESR
IBM Hardware
Export
http://127.0.0.1:4800/isav31/displayExport.do

Export Collector

Select a product: WebSphere Portal 6.0

Output directory:

Output file name (*.jar):

Figure A-13 Create the Portable Collector

- Once the Portable Collector has been created and moved to the WebSphere Portal Server machine, you simply extract the jar and execute the startcollector script. This script runs in console mode. Simply step through the text inputs to execute the desired collection type.

> **Note:** Review the list of collection scripts and choose the one that most fits the nature of your problem. Each collection is custom designed for a specific scenario.
>
> If you are unsure, choose **Portal General Problem**. If additional collection scripts are needed, the Level 2 support engineer who takes ownership of your PMR will provide further instructions on the specific script to run to collect the logs.

- Once the log collection is complete, move the zip file from the remote WebSphere Portal Server machine locally to the ISA machine.

- Once the logs are local, use the **login to ESR** link to launch the ESR tool to create a new PMR. During this process, you will be allowed to attach the log collection to the PMR. By doing this task, the logs will be made available to the support team at the time the PMR is opened.

For further details about the individual features offered by ISA, refer to the document, “The Support Authority: Getting help from the IBM Support Assistant”. The particularly useful guide can be found at:

http://www.ibm.com/developerworks/websphere/techjournal/0706_supauth/0706_supauth.html

IBM support site

The WebSphere Portal Server support site is the backbone of the IBM customer self-help tools. The IBM support sites are designed to be the main Web resource for support issues for a given product.

How does the support site help

IBM software solutions provide and maintain a series of Web pages that are designed to offer information, guidance, and direction to interested readers. Administrators and users of IBM WebSphere Portal are encouraged to visit and monitor the product support page for not only the portal product itself but for all the supporting software as well (operating systems, Application Server, Web server, database, and others) that might be used in your environment. Familiarity with the product support pages is central to maintaining a robust and well performing portal.

Examples of information presented include critical updates, scheduled maintenance, tips and hints, and a central location for other Self-Help initiatives available from IBM.

How can I access the support site

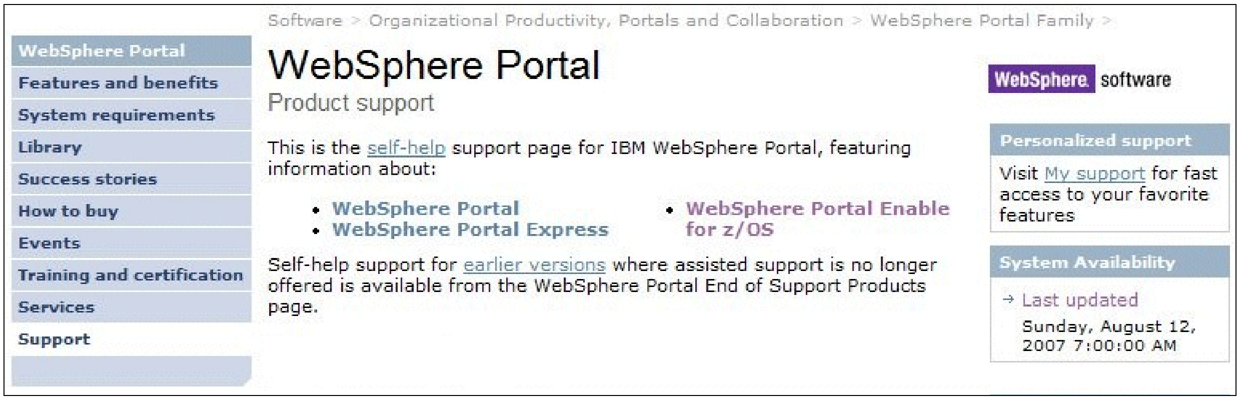
IBM WebSphere Portal’s support page is found at:

http://www.ibm.com/software/genservers/portal/support/

Best practices

IBM WebSphere Portal’s support page is arranged using the same format as other IBM software products: highlighted information near the top, core information in the center, and related information surrounding it on either side. The information surrounding the content occasionally changes in its presentation, so any static screen captures included in this section are subject to be outdated following the publication of this Redpaper; however, the key sections will remain available.

Consider the arrangement of information on the support page. The topmost sections of the page begin with a left navigation field, which highlights particular areas of interest, an introductory section, and a right column usually containing general IBM support information. Across the top is the familiar breadcrumb trail that is useful in navigating through the layers of IBM Web pages, as shown in Figure A-14.



Software > Organizational Productivity, Portals and Collaboration > WebSphere Portal Family >

- WebSphere Portal
- Features and benefits
- System requirements
- Library
- Success stories
- How to buy
- Events
- Training and certification
- Services
- Support

WebSphere Portal

Product support

WebSphere software

This is the self-help support page for IBM WebSphere Portal, featuring information about:

- WebSphere Portal
- WebSphere Portal Express
- WebSphere Portal Enable for z/OS

Self-help support for earlier versions where assisted support is no longer offered is available from the WebSphere Portal End of Support Products page.

Personalized support

Visit My support for fast access to your favorite features

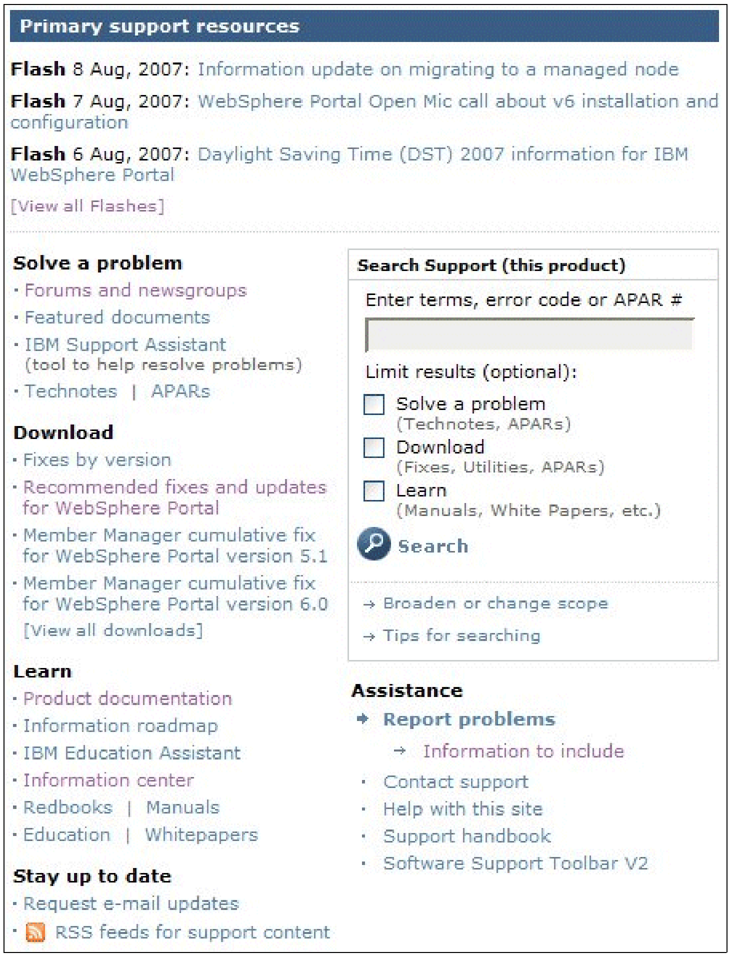
System Availability

→ Last updated

Sunday, August 12, 2007 7:00:00 AM

Figure A-14 Top of the WebSphere Portal product support page

Here our focus is on the central portion of the page, the Primary Support Resources section. See Figure A-15.



Primary support resources

Flash 8 Aug, 2007: Information update on migrating to a managed node

Flash 7 Aug, 2007: WebSphere Portal Open Mic call about v6 installation and configuration

Flash 6 Aug, 2007: Daylight Saving Time (DST) 2007 information for IBM WebSphere Portal

[View all Flashes]

Solve a problem

- Forums and newsgroups
- Featured documents
- IBM Support Assistant (tool to help resolve problems)
- Technotes | APARs

Download

- Fixes by version
- Recommended fixes and updates for WebSphere Portal
- Member Manager cumulative fix for WebSphere Portal version 5.1
- Member Manager cumulative fix for WebSphere Portal version 6.0

[View all downloads]

Learn

- Product documentation
- Information roadmap
- IBM Education Assistant
- Information center
- Redbooks | Manuals
- Education | Whitepapers

Stay up to date

- Request e-mail updates
- RSS feeds for support content

Search Support (this product)

Enter terms, error code or APAR #

Limit results (optional):

□ Solve a problem (Technotes, APARs)

□ Download (Fixes, Utilities, APARs)

□ Learn (Manuals, White Papers, etc.)

Search

→ Broaden or change scope

→ Tips for searching

Assistance

- Report problems
  - → Information to include
- Contact support
- Help with this site
- Support handbook
- Software Support Toolbar V2

Figure A-15 Primary Resources Section of the WebSphere Portal product support page

From the top, readers will see Flashes or News items that have been published and are of critical importance or otherwise should be brought to all visitors' attention. Examples are shown above, and have also been used to share information about critical code defects or updates that could prevent an interruption or outage for your portal.

The Solve a problem section is a collection of hyperlinks that focus on several information types. These are usually rather general and not specific to a particular release or edition of the software, but provide an excellent resource for the reader who knows what type of Self-Help resource they seek.

The Downloads section varies by product, and WebSphere Portal support teams have chosen to highlight these two documents, permanently anchored on the support page:

- Fixes by version: This link takes the visitor to the most current list of fixes available or fixes integrated into the most current maintenance release, identified by version.
- Recommended fixes and updates for WebSphere Portal: This link opens the most current specific set of recommendations for the Fix Pack and interim fix levels suggested for the majority of our customers. The page is designed to cover all currently supported releases in an easy-to-read and follow format, with links to any suggested downloads.

The remaining highlighted downloads found on the support page are the most recently published and available downloads, without regard to importance, edition, or version. Future updates to the support page could include other anchored links and are subject to change over time.

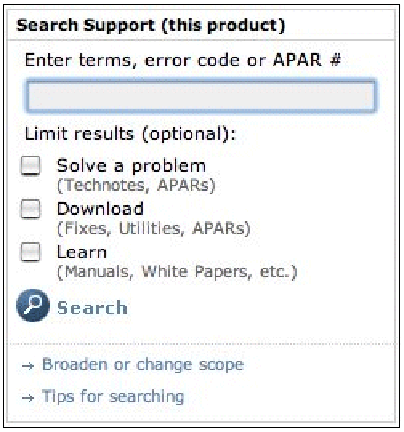
- The [View all downloads] link presents a list of all available downloads with the most recent added content at the top.

  Visitors seeking more in-depth information will find the links in the Learn section particularly useful. These links are also permanently anchored but will present the current information when followed.
- The Product documentation link opens the official product documentation page in the IBM developerWorks Portal Zone. Visitors will find the Information Centers for all current and previous portal releases here, plus additional documentation that provides guidance and detailed information regarding a variety of topics. Using a similar breadcrumb trail, you can go up a level in the hierarchy and land in the Portal Zone's main page to find even more white papers and other extensive documentation.
- The Information roadmap link takes you to the WebSphere Education page where you can learn more about the educational opportunities for WebSphere Portal and other associated IBM software solutions.
- The link to the IBM Education Assistant (a companion to the IBM Support and Guided Activity Assistants) offers collections of education modules and materials useful for many levels of experience. These offer an excellent way to get some familiarity with new technologies and products available from IBM.

The remaining links provide specifically constructed Search URLs that present visitors with the unique document types described in the links' titles.

The Stay up to date section offers links to subscribing to My Support (where you can define your specific areas of interest by technology, product, or offering) and receive periodic e-mail updates of new and updated Self-Help information from WebSphere Portal and other IBM solutions. You will also find the URL for the RSS feed of the WebSphere Portal support page, so that you can configure your RSS reader to receive timely notifications of recent additions.

Perhaps the single most valuable tool on the support page is the Search box. See Figure A-16.



Search Support (this product)

Enter terms, error code or APAR #

Limit results (optional):

□ Solve a problem (Technotes, APARs)

□ Download (Fixes, Utilities, APARs)

□ Learn (Manuals, White Papers, etc.)

Figure A-16 The WebSphere Portal product support page's search box

This is the primary input mechanism for finding the answers you need to the questions you might have. For example, if you are receiving a particular message code during some operations of your portal, copy that code into the search box and view the results. This is our interface to numerous datasources across the IBM sites and is the easiest way to use the page to meet your needs. Search results are provided with an indication of relevancy and even by date.

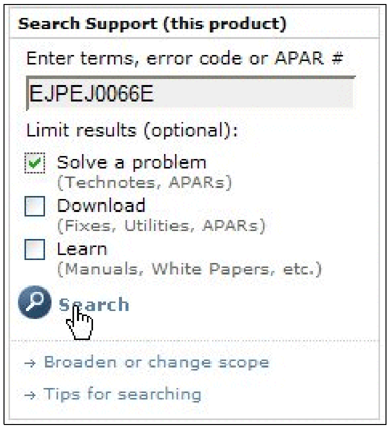
For this example, we have deployed our portal and are using custom themes and skins throughout. After assigning a new theme to the portal's Administration pages, the administrator has been unable to "assign access to a portlet" when using the Manage Portlets administrative portlet. Instead, the administrator receives the dreaded "This portlet is not available" message in its place.

Checking the wps_<date-time>.log file, the administrator sees this entry:

```
2007.08.16 19:03:22.101 E com.ibm.wps.engine.tags.PortletRenderTag doStartTag
EJPEJ0066E: The portlet could not be rendered. - StackTrace follows...
2007.08.16 19:03:22.101 E com.ibm.wps.engine.tags.PortletRenderTag
doStartTag
javax.portlet.PortletException
. . . .
Caused by: java.lang.NullPointerException
at
com.ibm.wps.portlets.permissions.PermissionsPortlet.doView(PermissionsPortlet.java
:192)…
```

The administrator also found that by restoring the default "out-of-the-box" theme for the admin pages, this problem did not occur.

To find the cause, the administrators visits the support page (http://www.ibm.com/software/genservers/portal/support/) and enters the message code into the Search box. See Figure A-17.



Search Support (this product)

Enter terms, error code or APAR #

EJPEJ0066E

Limit results (optional):

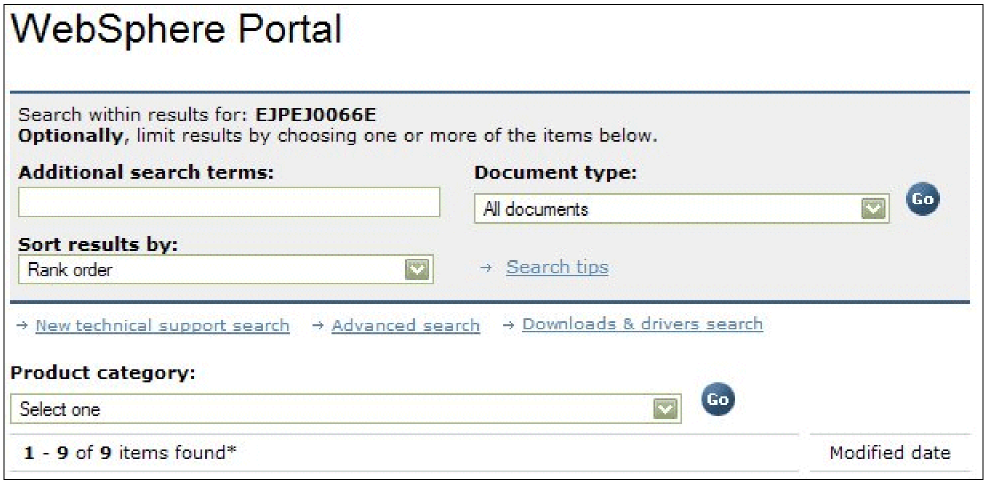
☑ Solve a problem (Technotes, APARs)

□ Download (Fixes, Utilities, APARs)

□ Learn (Manuals, White Papers, etc.)

Figure A-17 Text entry into the search box

The results page looks like Figure A-18.



WebSphere Portal

Search within results for: **EJPEJ0066E**
Optionally, limit results by choosing one or more of the items below.

Additional search terms:

Document type: All documents

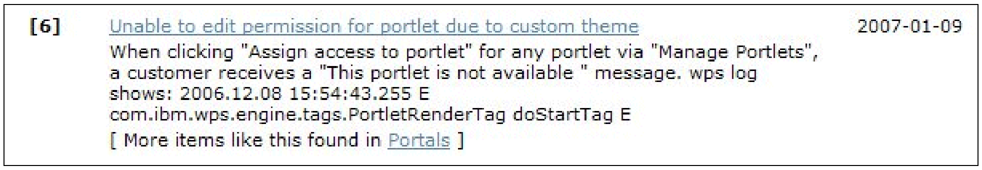
Sort results by: Rank order

Product category: Select one

1 - **9** of **9** items found*

Figure A-18 Results of the search

One of the results (number 6 at the time of this writing) is shown in Figure A-19.



[6] Unable to edit permission for portlet due to custom theme 2007-01-09

When clicking "Assign access to portlet" for any portlet via "Manage Portlets", a customer receives a "This portlet is not available " message. wps log shows: 2006.12.08 15:54:43.255 E com.ibm.wps.engine.tags.PortletRenderTag doStartTag E

[More items like this found in Portals]

Figure A-19 The answer

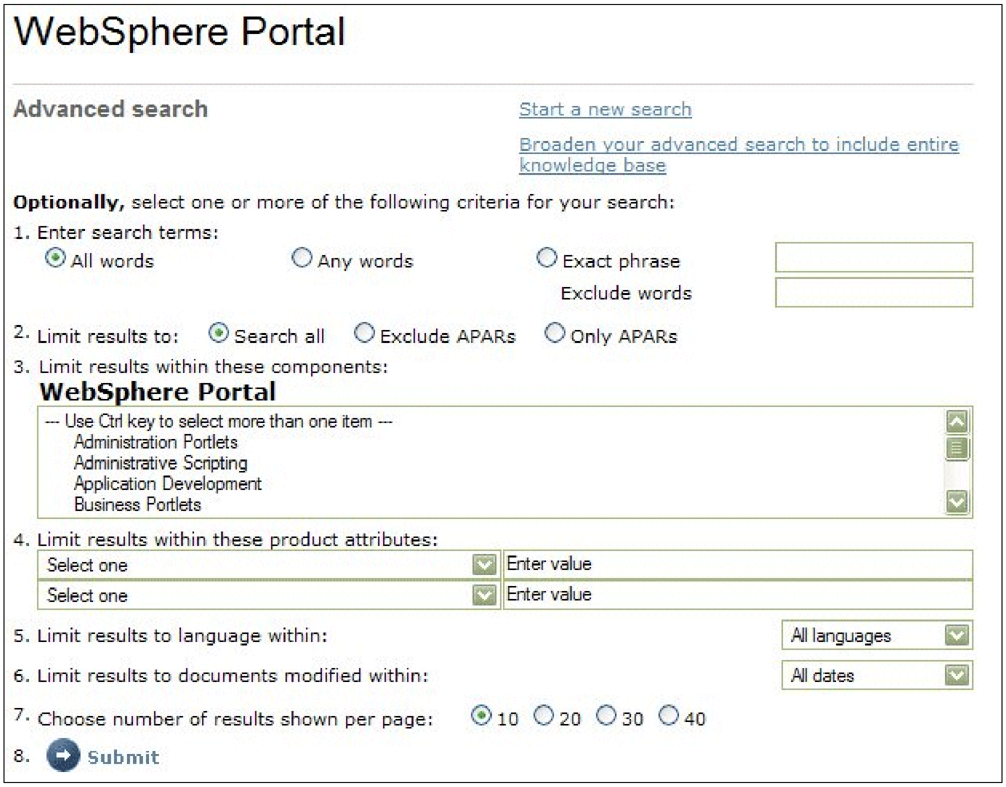
By following the link to that TechNote, the administrator finds the answer to the problem.

From the results page (Figure A-18), one can reach the Advanced search page, which is especially useful when one needs to discriminate the results by version, edition, platform, or other means.

The URL for our Advanced search page is simply:

http://www.ibm.com/support/advsrch.wss?rs=688

The Advanced search page for WebSphere Portal can be seen in Figure A-20.



WebSphere Portal

Advanced search

Start a new search

Broaden your advanced search to include entire knowledge base

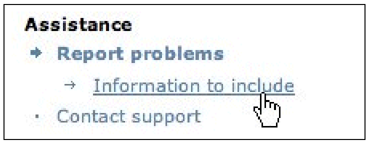
Optionally, select one or more of the following criteria for your search:

1. Enter search terms:
   ● All words ○ Any words ○ Exact phrase
   Exclude words
2. Limit results to: ● Search all ○ Exclude APARs ○ Only APARs
3. Limit results within these components:
   WebSphere Portal
   --- Use Ctrl key to select more than one item ---
   Administration Portlets
   Administrative Scripting
   Application Development
   Business Portlets
4. Limit results within these product attributes:
   Select one | Enter value
   Select one | Enter value
5. Limit results to language within: All languages
6. Limit results to documents modified within: All dates
7. Choose number of results shown per page: ● 10 ○ 20 ○ 30 ○ 40
8. Submit

Figure A-20 Advanced search for WebSphere Portal

Each of the items above is self-explanatory and can be used to tailor your results to meet your needs. For more information describing the components shown in number 3's list above, refer to the TechNote "Explanation of Functional Areas and Components of IBM WebSphere Portal and WebSphere Portal Express, version 6.0" (http://www.ibm.com/support/docview.wss?rs=688&uid=swg27009175). Understanding these components and functional areas will help you identify the area that concerns you.

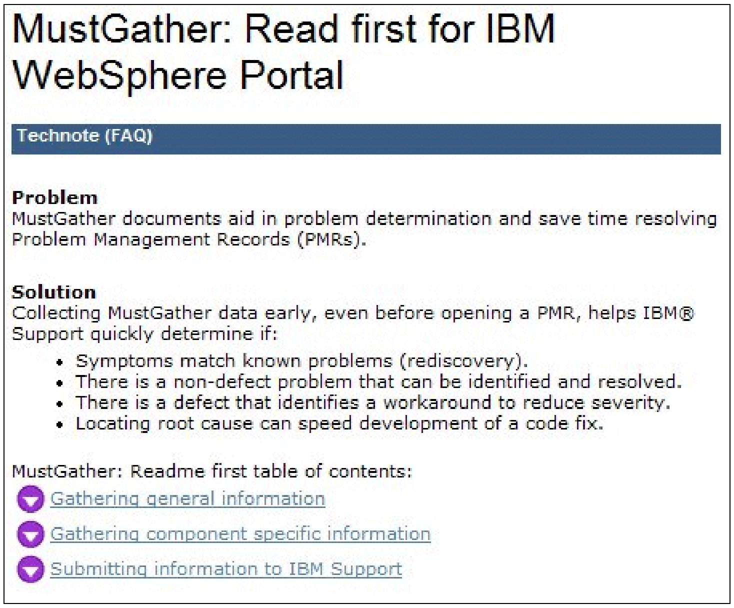
If you need more help after exploring the support page, continue to the Assistance section. Another of our valuable Self-Help tools can be found there. See Figure A-21.



Assistance
- **Report problems**
  - Information to include
- Contact support

Figure A-21 Find the MustGather documents under "Information to include"

Though originally designed as guidance for Directed Help for customers opening problem tickets (PMRs), the intrepid Do It Yourself enthusiast can use these documents to help isolate the problematic area for a variety of problems, get the relevant diagnostic data to troubleshoot it, and resolve the situation without ever having to call IBM or open a PMR. The Information to include link opens up the TechNote "MustGather: Read first for IBM WebSphere Portal" (`http://www.ibm.com/support/docview.wss?rs=688&uid=swg21236371`), as shown in Figure A-22.



MustGather: Read first for IBM WebSphere Portal

Technote (FAQ)

Problem
MustGather documents aid in problem determination and save time resolving Problem Management Records (PMRs).

Solution
Collecting MustGather data early, even before opening a PMR, helps IBM® Support quickly determine if:

- Symptoms match known problems (rediscovery).
- There is a non-defect problem that can be identified and resolved.
- There is a defect that identifies a workaround to reduce severity.
- Locating root cause can speed development of a code fix.

MustGather: Readme first table of contents:

Gathering general information

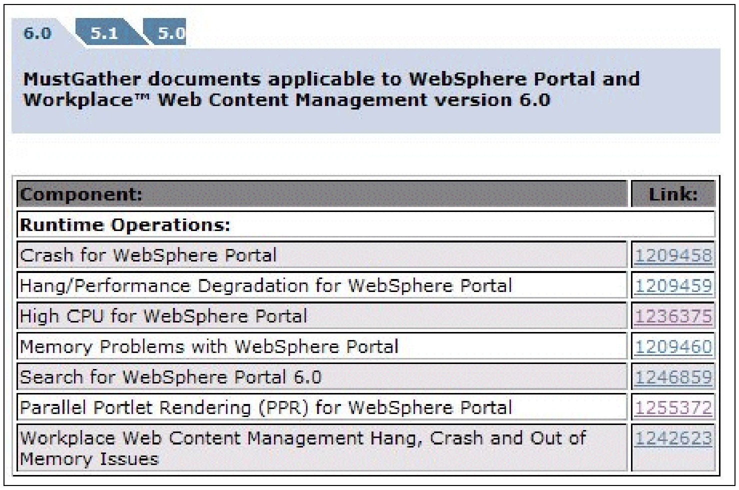
Gathering component specific information

Submitting information to IBM Support

Figure A-22 MustGather: Read first page

This page is currently available for Versions 6.0, 5.1 and 5.0. Future releases will be added as they become available. The term "MustGather" refers to the information absolutely necessary to help identify the majority of problems encountered for a given function or aspect of the portal. Customers willing to invest their time in learning how to troubleshoot their own issues find these MustGather documents an invaluable tool in the Self-Help arsenal.

The component-specific information helps the reader identify the function or action under investigation, such as the Runtime problems shown in Figure A-23.



MustGather documents applicable to WebSphere Portal and Workplace™ Web Content Management version 6.0

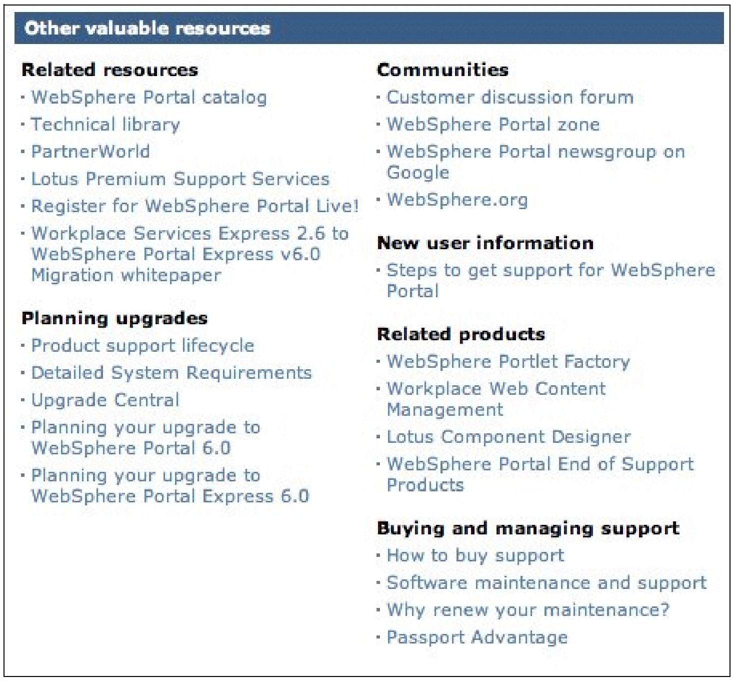
Component:	Link:
Runtime Operations:	
Crash for WebSphere Portal	1209458
Hang/Performance Degradation for WebSphere Portal	1209459
High CPU for WebSphere Portal	1236375
Memory Problems with WebSphere Portal	1209460
Search for WebSphere Portal 6.0	1246859
Parallel Portlet Rendering (PPR) for WebSphere Portal	1255372
Workplace Web Content Management Hang, Crash and Out of Memory Issues	1242623

Figure A-23 MustGather: Read first – versions and components

Other areas covered include Installation/Configuration, Security/Administration, Content/Document Management, and Portlets/Development/Customization.

Many of the diagnostic collections of data covered in these MustGather documents have also been automated by cross-platform scripts that run under the AutoPD tool, which is covered in "Use case examples - Service" on page 184. See each individual MustGather document for more details.

The last topic to cover regarding the support page itself is the lower middle section, Other valuable resources are shown in Figure A-24.



Other valuable resources

Related resources
- WebSphere Portal catalog
- Technical library
- PartnerWorld
- Lotus Premium Support Services
- Register for WebSphere Portal Live!
- Workplace Services Express 2.6 to WebSphere Portal Express v6.0 Migration whitepaper

Planning upgrades
- Product support lifecycle
- Detailed System Requirements
- Upgrade Central
- Planning your upgrade to WebSphere Portal 6.0
- Planning your upgrade to WebSphere Portal Express 6.0

Communities
- Customer discussion forum
- WebSphere Portal zone
- WebSphere Portal newsgroup on Google
- WebSphere.org

New user information
- Steps to get support for WebSphere Portal

Related products
- WebSphere Portlet Factory
- Workplace Web Content Management
- Lotus Component Designer
- WebSphere Portal End of Support Products

Buying and managing support
- How to buy support
- Software maintenance and support
- Why renew your maintenance?
- Passport Advantage

Figure A-24 Other valuable resources

These links found in this section are of particular importance to Self-Help interests:

- WebSphere Portal catalog: A collection of portlets provided by IBM and other vendors, offering solutions to business integration needs, technology introductions, and example/reference portlets to use in your environment. Some are free, some are limited use, and others are available for charge only.
- Product support life cycle: This page lists the various releases of the WebSphere Portal family of products, their dates of introduction and, if available, their End of Service (EOS) dates.
- Detailed System Requirements: Find the supported and minimum levels of operating systems, Application Servers, Web servers, databases, and other applications used in conjunction with the portal. These documents are listed by version and release.
- Upgrade Central: A very useful knowledge collection of documents related to Migration (such as moving from Version 5.1 to 6.0) and Upgrades (such as installing Fix Pack 6.0.1.1 into Version 6.0).
- Customer discussion forum: Link to the developerWorks interface to the ibm.software.websphere.portal-server Usenet newsgroup.
- WebSphere Portal zone: Also hosted on developerWorks, this collection of information provides in-depth articles on all matters related to installing, deploying, administering, and especially developing content for your portal. This page should definitely be bookmarked.

Overall, the best practice is to take time to familiarize yourself with the resources available on the product support page, especially those discussed above. Gaining experience with the page and how to use it to your advantage before a crisis erupts will allow you to know where to look and how to find the answers you need, without having to learn on the fly.

Read through the MustGather documents, or at least a couple of them, to understand the methodology described in the documents to troubleshoot your own problems, and save valuable time by knowing what to look for and where.

Subscribe to MySupport and the RSS feeds for the product support pages of applications you have deployed in conjunction with your portal.

Take advantage of the free resources (IBM Support, Guided Activity and Education Assistants, and the Software Support Toolbar) to learn and guide you in your portal administration and usage.

Use the WebSphere Portal user forum to your advantage. Representatives from IBM development, support, and test routinely monitor and contribute to the forums, as do customers who might have encountered (and found answers to) your similar questions.

Study the Table of Contents for the Information Centers and abstracts for white papers and highlighted TechNotes. It is often more useful to remember where you have seen some information for future reference rather than the complete contents of the documentation. Bookmark the pages you find most useful.

IBM online communities

Participation in WebSphere Portal Server communities, such as newsgroups, forums, wikis, blogs, and podcasts, is strongly encouraged and highly recommended.

How do online communities help

The collective knowledge sharing and experience of online communities is a very powerful resource for problem determination and problem solving. Participating in online communities is also beneficial in problem avoidance. Online communities are very valuable when you need to obtain direction and product best practices. By leveraging the vast experience of the collective community, one can avoid known pitfalls and get started down the correct path from the outset of a project.

How can I access the online communities

Complete information about WebSphere Portal Server communities can be found at the developerWorks site at this link:

http://www.ibm.com/developerworks/community/

Bookmark this page, as it should serve as your main entry point into the world of IBM online community resources.

To get started using IBM communities, begin with the page “New to developerWorks Community”. This page can be linked to from the main community page:

http://www.ibm.com/developerworks/community/newto/

From this page, you can link to the “Get Started” information for blogs, forums, podcasts, and wikis.

Best practices

One of the most powerful community resources is the WebSphere Portal Server newsgroup and forum. The WebSphere Portal Server product team monitors the forum located at ibm.software.websphere.portal-server. We recommend, as a best practice, that you actively participate in this newsgroup and forum on a regular basis.

Quick access to newsgroups and forums can be linked to from the ISA tool through the Product Information feature, as described “IBM Support Assistant (ISA)” on page 170.

> **Important:** WebSphere Portal Server is built on top of WebSphere Application Server, so a knowledge of WebSphere Application Server troubleshooting is critical as well. So it is also highly recommended to take advantage of, and participate in, the WebSphere Application Server forums as well.

IBM RSS feeds

RSS is a quick, easy, and lightweight format for monitoring new content added to Web sites. An RSS feed uses XML-formatted files to deliver content that you then access with an RSS Reader. Our RSS feeds provide the title of a new piece of content, such as a TechNote, a description of the new content, and a link to the content. Feeds are updated daily.

The WebSphere Portal Server RSS feed is a great way to receive the most current news and technical updates about WebSphere Portal Server.

How do RSS feeds help

The best way to state the value of RSS feeds is that they keep you informed and connected. The WebSphere Portal Server product teams leverage the RSS feeds on a regular basis to push important technical articles about WebSphere Portal Server to customers and the user community. RSS feeds can be customized to only push the content that is important to you and your environment.

How can I access the IBM RSS feeds

Typically, a RSS reader is used to obtain and manage the RSS feeds.

Use the following link as a step-by-step guide to understanding how to introduce RSS into your environment:

`http://www-306.ibm.com/software/support/rsshelp.html`

To learn more about the RSS feeds available from IBM, access the IBM developerWorks site at:

`http://www.ibm.com/developerworks/rss/`

Best practices

For best practice approaches to using IBM RSS feeds to promote self-help, refer to the following article, “Introduction to Syndication, (RSS) Really Simple Syndication”, found at:

`http://www.ibm.com/developerworks/xml/library/x-rssintro/index.html`

> **Important:** WebSphere Portal Server is built on top of WebSphere Application Server, so a knowledge of WebSphere Application Server troubleshooting is critical as well. So we also highly recommend taking advantage of the WebSphere Application Server RSS feeds as well.

IBM Support Toolbar

The IBM Support Toolbar offers quick access to many of IBM support’s sites and tools from one well organized toolbar.

How does the IBM Support Toolbar help

The IBM Support Toolbar offers several benefits, and the largest benefit you will notice is saved time. It is one place that leads you to the most accessed supported pages regardless of what IBM products you are using. It allows you to quickly search your choice of content residing on several of IBM's servers using the search capabilities from wherever you are on the Web. It is fast, convenient, and provides results.

How can I obtain the IBM Support Toolbar

You can find the toolbar at:

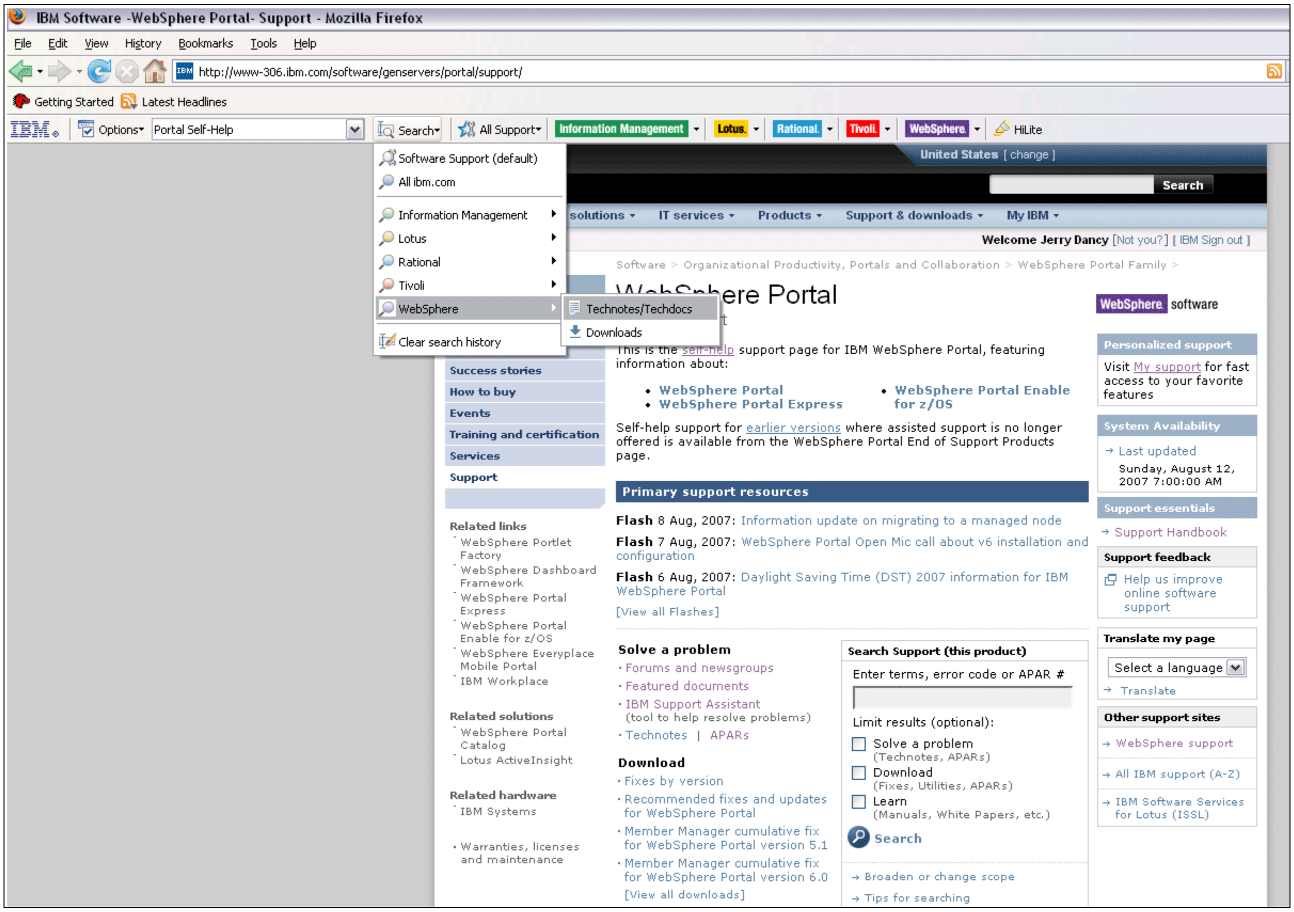
`http://www-306.ibm.com/software/support/toolbar/index.html?ibmsst=ibmTbMenu`

IBM Support Toolbar buttons

Here we discuss the various buttons on the toolbar and their functions.

Search button

Enter the desired search string directly into the text box on the toolbar and then click the **Search** button to search across all of IBM support, or narrow it to a specific product, as shown in Figure A-25.


IBM Software -WebSphere Portal- Support - Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://www-306.ibm.com/software/genservers/portal/support/
Getting Started Latest Headlines
IBM. Options Portal Self-Help Search All Support Information Management Lotus Rational Tivoli WebSphere HiLite
Software Support (default)
All ibm.com
Information Management
Lotus
Rational
Tivoli
WebSphere
Technotes/Techdocs
Downloads
Clear search history
United States [change]
Search
solutions IT services Products Support & downloads My IBM
Welcome Jerry Dancy [Not you?] [IBM Sign out]
Software > Organizational Productivity, Portals and Collaboration > WebSphere Portal Family >
WebSphere Portal
WebSphere software
Success stories
How to buy
Events
Training and certification
Services
Support
Related links
WebSphere Portlet Factory
WebSphere Dashboard Framework
WebSphere Portal Express
WebSphere Portal Enable for z/OS
WebSphere Everyplace Mobile Portal
IBM Workplace
Related solutions
WebSphere Portal Catalog
Lotus ActiveInsight
Related hardware
IBM Systems
Warranties, licenses and maintenance
Personalized support
System Availability
Support essentials
Support Handbook
Support feedback
Help us improve online software support
Translate my page
Select a language
Translate
Other support sites
WebSphere support
All IBM support (A-Z)
IBM Software Services for Lotus (ISSL)
Primary support resources
Solve a problem
Forums and newsgroups
Featured documents
IBM Support Assistant (tool to help resolve problems)
Technotes | APARs
Download
Fixes by version
Recommended fixes and updates for WebSphere Portal
Member Manager cumulative fix for WebSphere Portal version 5.1
Member Manager cumulative fix for WebSphere Portal version 6.0
[View all downloads]
Search Support (this product)
Search
Broaden or change scope
Tips for searching

This is the self-help support page for IBM WebSphere Portal, featuring information about:

- WebSphere Portal
- WebSphere Portal Express
- WebSphere Portal Enable for z/OS

Self-help support for earlier versions where assisted support is no longer offered is available from the WebSphere Portal End of Support Products page.

Visit My support for fast access to your favorite features

Last updated Sunday, August 12, 2007 7:00:00 AM

Flash 8 Aug, 2007: Information update on migrating to a managed node

Flash 7 Aug, 2007: WebSphere Portal Open Mic call about v6 installation and configuration

Flash 6 Aug, 2007: Daylight Saving Time (DST) 2007 information for IBM WebSphere Portal

[View all Flashes]

Enter terms, error code or APAR #

Limit results (optional):

□ Solve a problem (Technotes, APARs)
□ Download (Fixes, Utilities, APARs)
□ Learn (Manuals, White Papers, etc.)

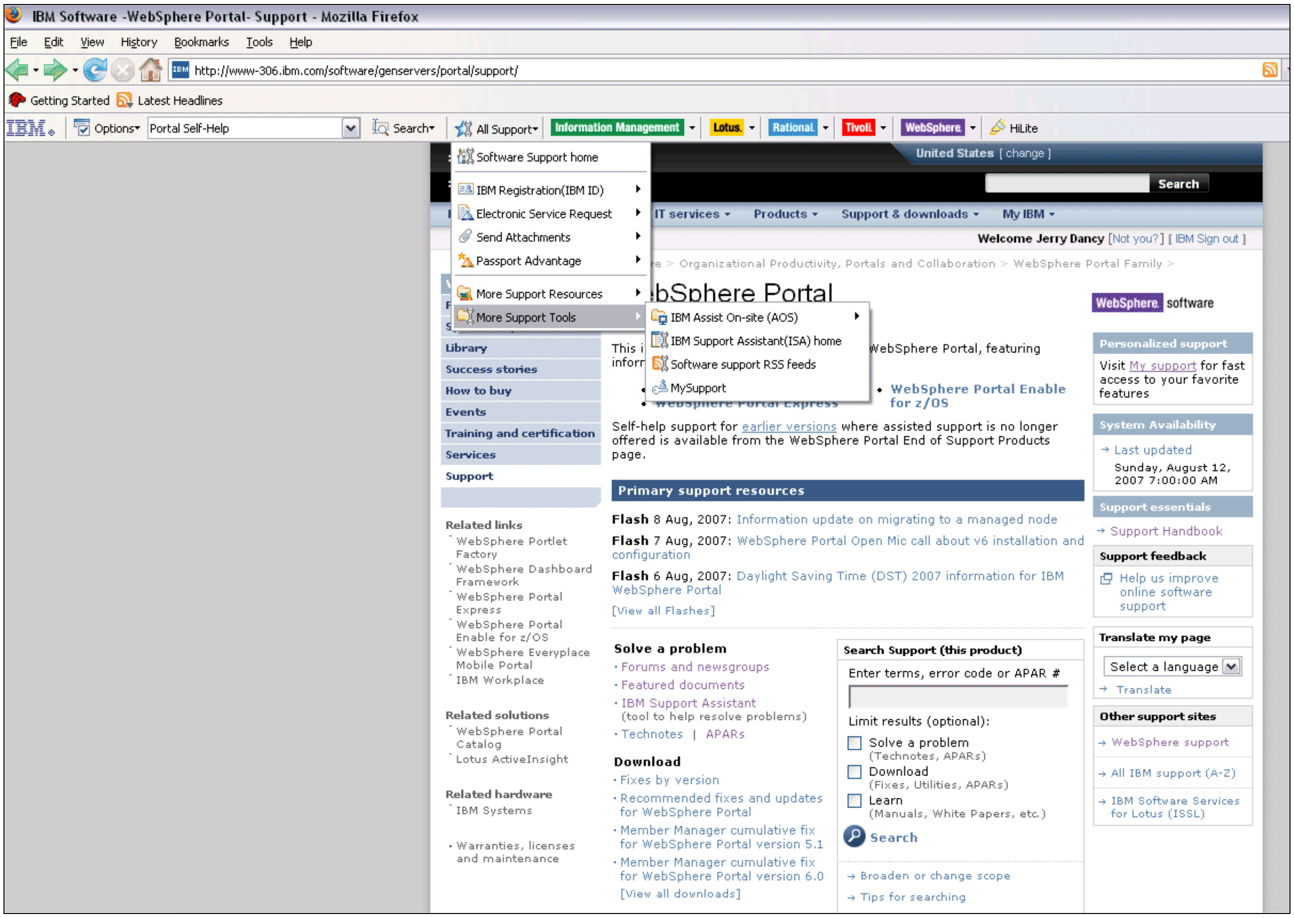
Figure A-25 Toolbar Search button

All Support button

The All Support button allows quick access to general IBM support tools, including:

- IBM ID registration: This tool is needed to access many IBM Web sites.
- Electronic Service Request (ESR): This tool is used to manage PMRs online.
- EcuRep: This tool is used to submit logs/files to IBM when working with support.
- Passport Advantage: This tool is used to obtain licensed IBM code.
- Assist on-site: This tool is used to allow IBM remote support access into your environment.
- IBM Support Assistant.
- Quick access to RSS feeds.
- Quick access to MySupport.

See Figure A-26.



bSphere Portal

This i... WebSphere Portal, featuring
infor...

- WebSphere Portal Express
- WebSphere Portal Enable for z/OS

Self-help support for earlier versions where assisted support is no longer offered is available from the WebSphere Portal End of Support Products page.

Primary support resources

Flash 8 Aug, 2007: Information update on migrating to a managed node

Flash 7 Aug, 2007: WebSphere Portal Open Mic call about v6 installation and configuration

Flash 6 Aug, 2007: Daylight Saving Time (DST) 2007 information for IBM WebSphere Portal

[View all Flashes]

Solve a problem
- Forums and newsgroups
- Featured documents
- IBM Support Assistant (tool to help resolve problems)
- Technotes | APARs

Download
- Fixes by version
- Recommended fixes and updates for WebSphere Portal
- Member Manager cumulative fix for WebSphere Portal version 5.1
- Member Manager cumulative fix for WebSphere Portal version 6.0

[View all downloads]

Search Support (this product)

Enter terms, error code or APAR #

Limit results (optional):

□ Solve a problem (Technotes, APARs)
□ Download (Fixes, Utilities, APARs)
□ Learn (Manuals, White Papers, etc.)

Search

→ Broaden or change scope
→ Tips for searching

Related links
- WebSphere Portlet Factory
- WebSphere Dashboard Framework
- WebSphere Portal Express
- WebSphere Portal Enable for z/OS
- WebSphere Everyplace Mobile Portal
- IBM Workplace

Related solutions
- WebSphere Portal Catalog
- Lotus ActiveInsight

Related hardware
- IBM Systems

- Warranties, licenses and maintenance

Personalized support

Visit My support for fast access to your favorite features

System Availability

→ Last updated Sunday, August 12, 2007 7:00:00 AM

Support essentials

→ Support Handbook

Support feedback

Help us improve online software support

Translate my page

Select a language

→ Translate

Other support sites

→ WebSphere support

→ All IBM support (A-Z)

→ IBM Software Services for Lotus (ISSL)

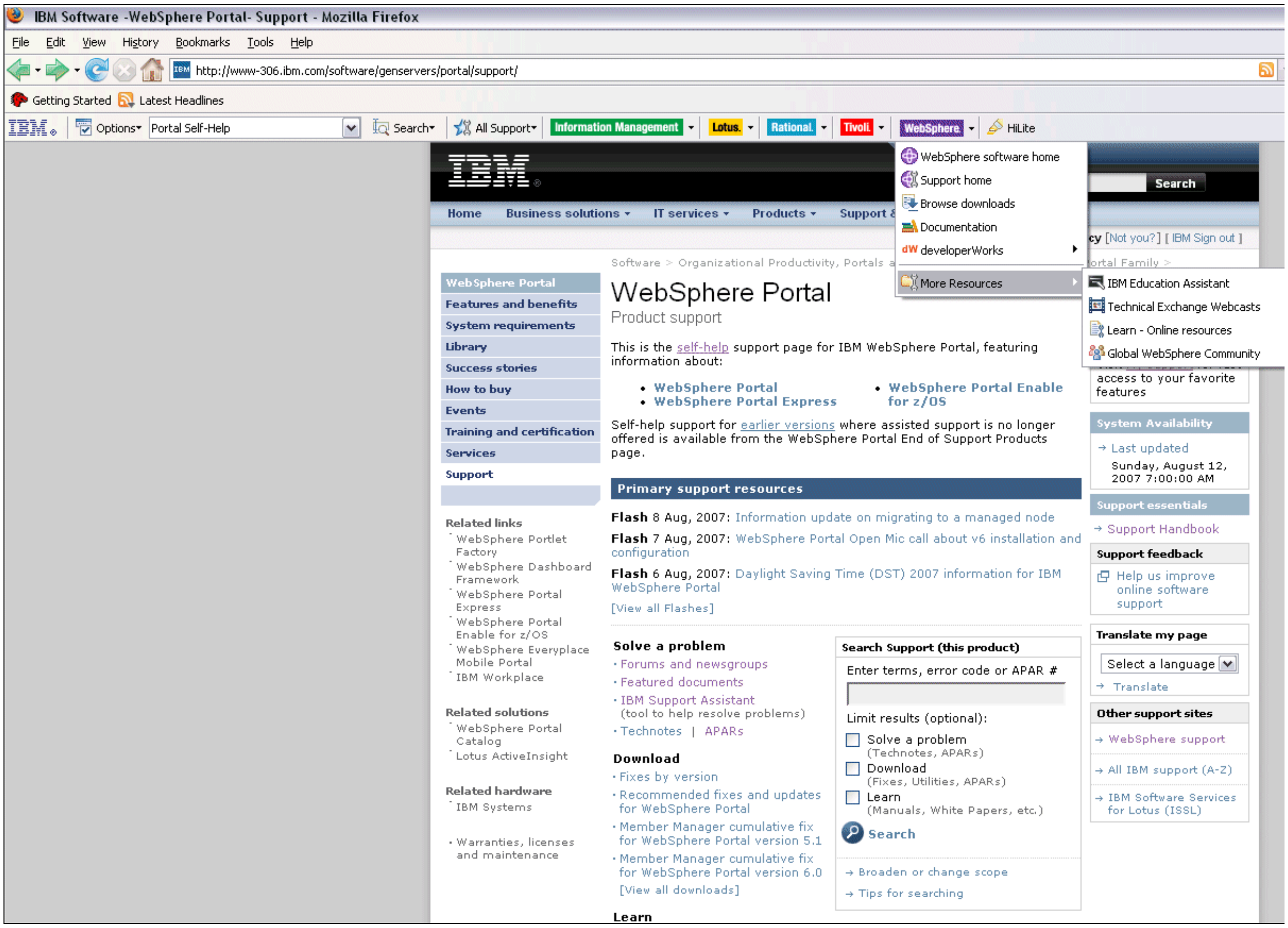
Figure A-26 Toolbar All Support button

WebSphere button

The WebSphere button allows quick access to product specific support tools, including:

- Quick access to product specific software and support pages
- Quick access to newsgroups and forums
- Quick access to training and certification roadmaps
- Quick access to the IBM Education Assistant and other learning resources

See Figure A-27.



WebSphere Portal

Product support

This is the self-help support page for IBM WebSphere Portal, featuring information about:

- WebSphere Portal
- WebSphere Portal Express
- WebSphere Portal Enable for z/OS

Self-help support for earlier versions where assisted support is no longer offered is available from the WebSphere Portal End of Support Products page.

Primary support resources

Flash 8 Aug, 2007: Information update on migrating to a managed node

Flash 7 Aug, 2007: WebSphere Portal Open Mic call about v6 installation and configuration

Flash 6 Aug, 2007: Daylight Saving Time (DST) 2007 information for IBM WebSphere Portal

[View all Flashes]

Solve a problem
Forums and newsgroups
Featured documents
IBM Support Assistant (tool to help resolve problems)
Technotes | APARs
Download
Fixes by version
Recommended fixes and updates for WebSphere Portal
Member Manager cumulative fix for WebSphere Portal version 5.1
Member Manager cumulative fix for WebSphere Portal version 6.0
[View all downloads]
Learn
Search Support (this product)
Enter terms, error code or APAR #
Limit results (optional):
□ Solve a problem (Technotes, APARs)
□ Download (Fixes, Utilities, APARs)
□ Learn (Manuals, White Papers, etc.)
Search
Broaden or change scope
Tips for searching
access to your favorite features
System Availability
Last updated
Sunday, August 12, 2007 7:00:00 AM
Support essentials
Support Handbook
Support feedback
Help us improve online software support
Translate my page
Select a language
Translate
Other support sites
WebSphere support
All IBM support (A-Z)
IBM Software Services for Lotus (ISSL)

Figure A-27 Toolbar WebSphere button

IBM Education Assistant

The IBM Education Assistant is a tool designed to provide guidance and instruction for various tasks or procedures.

How does the IBM Education Assistant help

IBM Education Assistant is a collection of multimedia educational modules designed to help you gain a better understanding of IBM software products and use them more effectively to meet your business requirements. Modules consist of the following types of content:

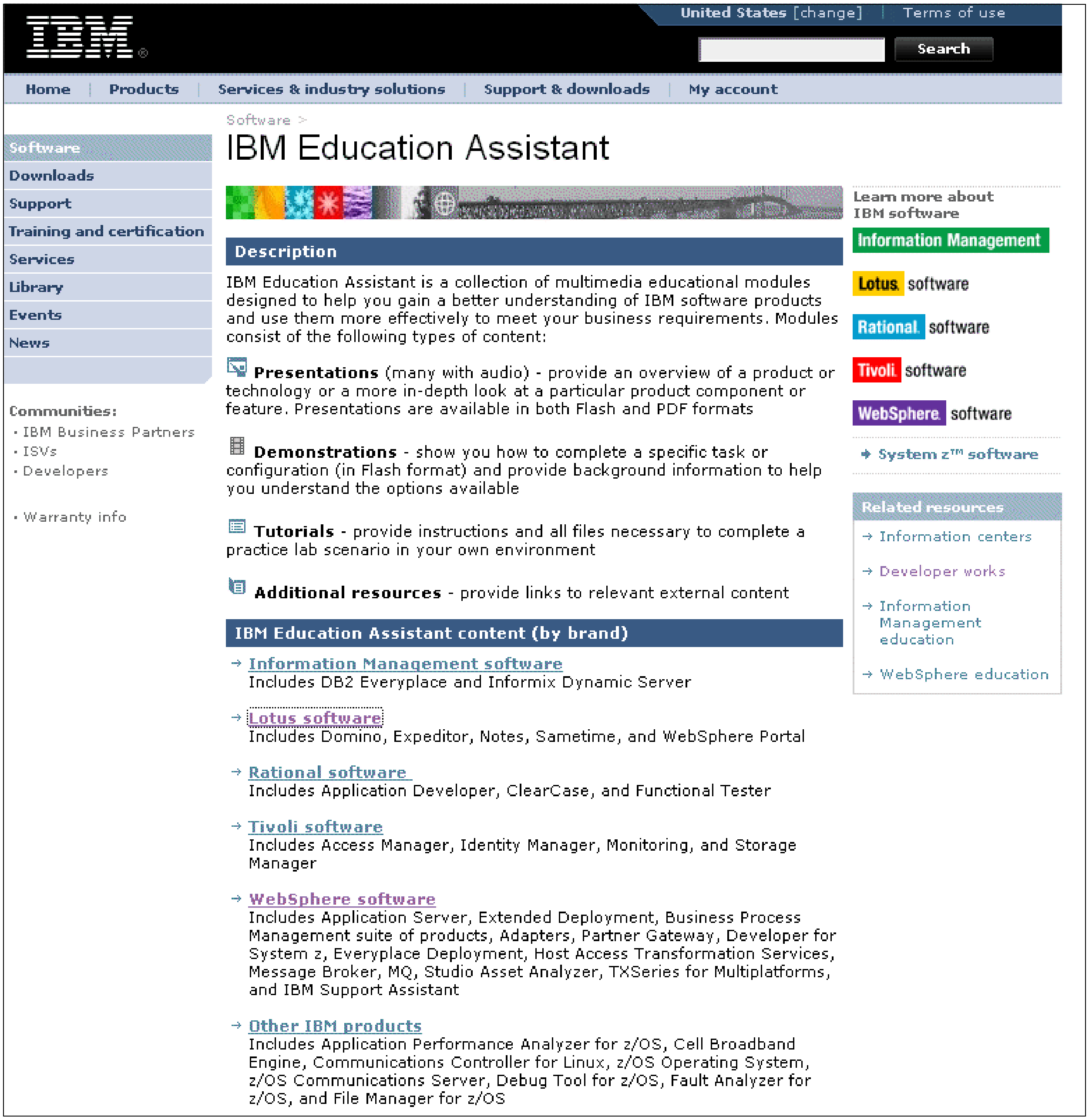
- Presentations (many with audio): Provide an overview of a product or technology or a more in-depth look at a particular product component or feature. Presentations are available in both Flash and PDF formats.
- Demonstrations: Show you how to complete a specific task or configuration (in Flash format) and provide background information to help you understand the options available.
- Tutorials: Provide instructions and all files necessary to complete a practice lab scenario in your own environment.
- Additional resources: Provide links to relevant external content.

How can I access the IBM Education Assistant

The IBM Education Assistant software page can be found at the following link:

http://www-306.ibm.com/software/info/education/assistant/

From this page, you can link to content by brand. See Figure A-28.



United States [change] | Terms of use
Search
Home | Products | Services & industry solutions | Support & downloads | My account
Software
Downloads
Support
Training and certification
Services
Library
Events
News
Communities:
· IBM Business Partners
· ISVs
· Developers
· Warranty info
Software >

IBM Education Assistant

Description

IBM Education Assistant is a collection of multimedia educational modules designed to help you gain a better understanding of IBM software products and use them more effectively to meet your business requirements. Modules consist of the following types of content:

Presentations (many with audio) - provide an overview of a product or technology or a more in-depth look at a particular product component or feature. Presentations are available in both Flash and PDF formats

Demonstrations - show you how to complete a specific task or configuration (in Flash format) and provide background information to help you understand the options available

Tutorials - provide instructions and all files necessary to complete a practice lab scenario in your own environment

Additional resources - provide links to relevant external content

IBM Education Assistant content (by brand)

→ Information Management software
Includes DB2 Everyplace and Informix Dynamic Server

→ Lotus software
Includes Domino, Expeditor, Notes, Sametime, and WebSphere Portal

→ Rational software
Includes Application Developer, ClearCase, and Functional Tester

→ Tivoli software
Includes Access Manager, Identity Manager, Monitoring, and Storage Manager

→ WebSphere software
Includes Application Server, Extended Deployment, Business Process Management suite of products, Adapters, Partner Gateway, Developer for System z, Everyplace Deployment, Host Access Transformation Services, Message Broker, MQ, Studio Asset Analyzer, TXSeries for Multiplatforms, and IBM Support Assistant

→ Other IBM products
Includes Application Performance Analyzer for z/OS, Cell Broadband Engine, Communications Controller for Linux, z/OS Operating System, z/OS Communications Server, Debug Tool for z/OS, Fault Analyzer for z/OS, and File Manager for z/OS

Figure A-28 IBM Education Assistant main page

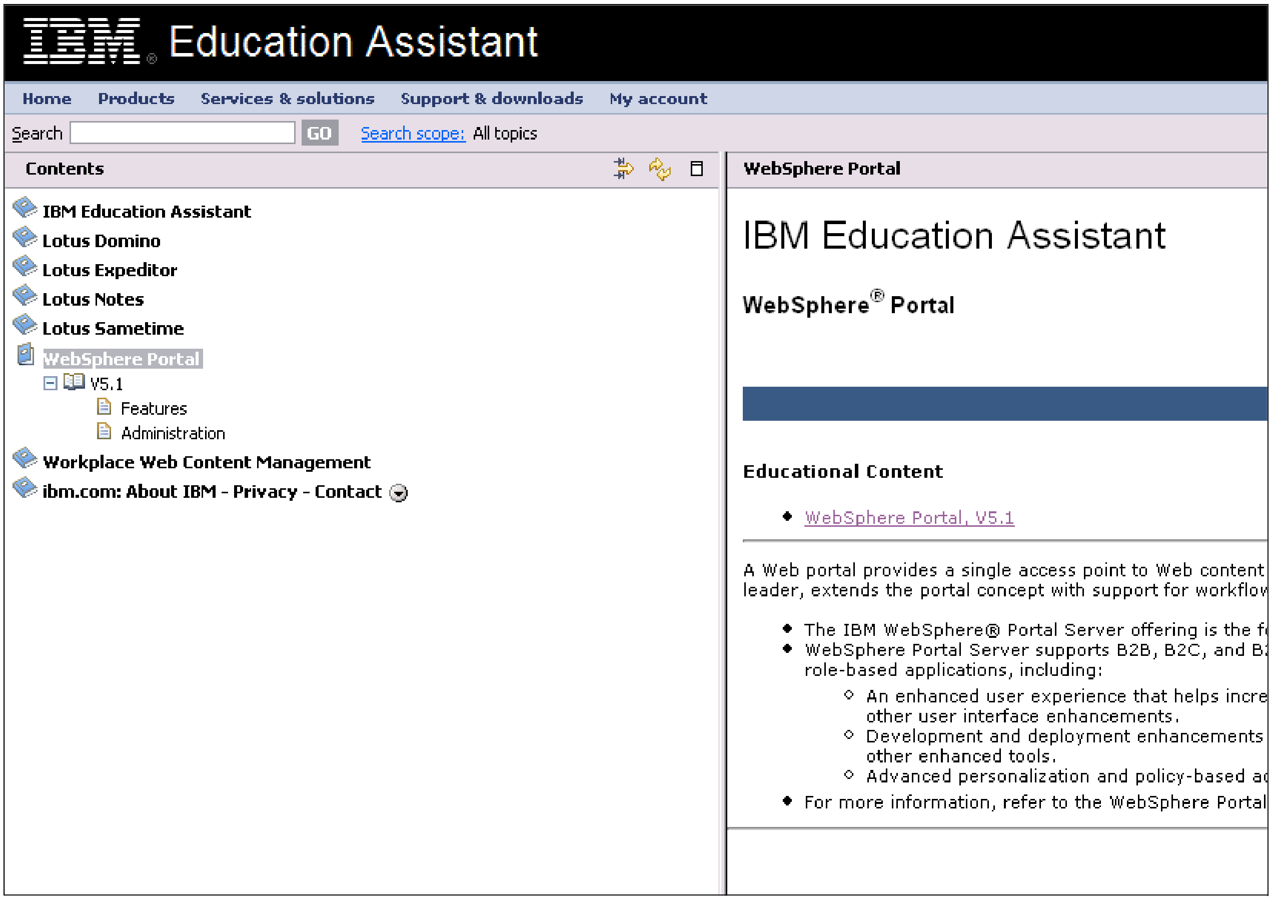
Best practices

When first beginning a new project, get into the habit of accessing the IBM Education Assistant to see if any content currently exists for the scenario or procedure you are about to perform.

The IBM Education Assistant is always being updated with new content, so check back regularly to see if any new content is available.

From the main page, click the **Lotus Software** link to access the WebSphere Portal Server specific content. See Figure A-29.

Important: WebSphere Portal Server is built on top of WebSphere Application Server, so a knowledge of WebSphere Application Server troubleshooting is critical as well. We highly recommend taking advantage of the WebSphere Application Server content as well.



IBM Education Assistant

WebSphere® Portal

Educational Content

- WebSphere Portal, V5.1

A Web portal provides a single access point to Web content
leader, extends the portal concept with support for workflov

- The IBM WebSphere® Portal Server offering is the f
- WebSphere Portal Server supports B2B, B2C, and B
role-based applications, including:
  - An enhanced user experience that helps incre
other user interface enhancements.
  - Development and deployment enhancements
other enhanced tools.
  - Advanced personalization and policy-based a
- For more information, refer to the WebSphere Portal

Figure A-29 IBM Education Assistant Portal content

IBM Guided Activity Assistant (IGAA)

The IBM Guided Activity Assistant (IGAA) is a new tool that brings together all three of these support elements (information, tools, and processes) to help you solve problems in an easier and more consistent manner. IGAA takes you step-by-step through the IBM-recommended problem determination process and exposes you to information, utilities, and best practices at the points where they are appropriate and relevant.

How does the IBM Guided Activity Assistant help

The IBM Guided Activity Assistant aims to help you answer the following questions:

- What should I do next?
- What diagnostic data should I analyze?
- What tool should I use to analyze it?
- How do I install and interact with that tool?

IGAA helps you at every step of a problem determination scenario.

If you are a new user and want to understand what step to take next, IGAA can explain to you in detail what next step is appropriate, and provide supporting information to give you a better understanding of the problem and the recommended solutions. IGAA does this with the philosophy of “guide first, educate later”. This means IGAA tries to expose you to enough information to help you work through the current problem, but not too much that it requires extensive reading just to accomplish the first step. After you have resolved your problem, you can then retrace your steps to educate yourself at a deeper level by following designated links to the appropriate IBM Redbooks publications, Information Center pages, IBM Education Assistant modules, and other resources.

If you are an advanced user and want a quick reminder of the problem determination steps to take, IGAA helps you by only showing the information necessary to move you quickly through the steps. At each point along the path, additional information is only a click away if you need specific details about any step in the problem determination workflow.

While the primary goal of IGAA is to guide you through the process of problem determination, IGAA also helps you stay organized as you work through the problem determination process. At times, you might need to collect diagnostic data for deeper analysis. Keeping these files organized and associated with your problem determination session helps to keep the process efficient. You can upload the files or link to them if they are too large to make copies; those files can then be passed seamlessly to the appropriate tools that are recommended in the flows.

If IGAA does not have a flow that helps you solve your problem, it can help you engage the appropriate IBM support representatives. IGAA will quickly bundle up the information you have already gathered and send that information up to IBM Support, who can then pick up right where you left off, saving a significant amount of information gathering time.

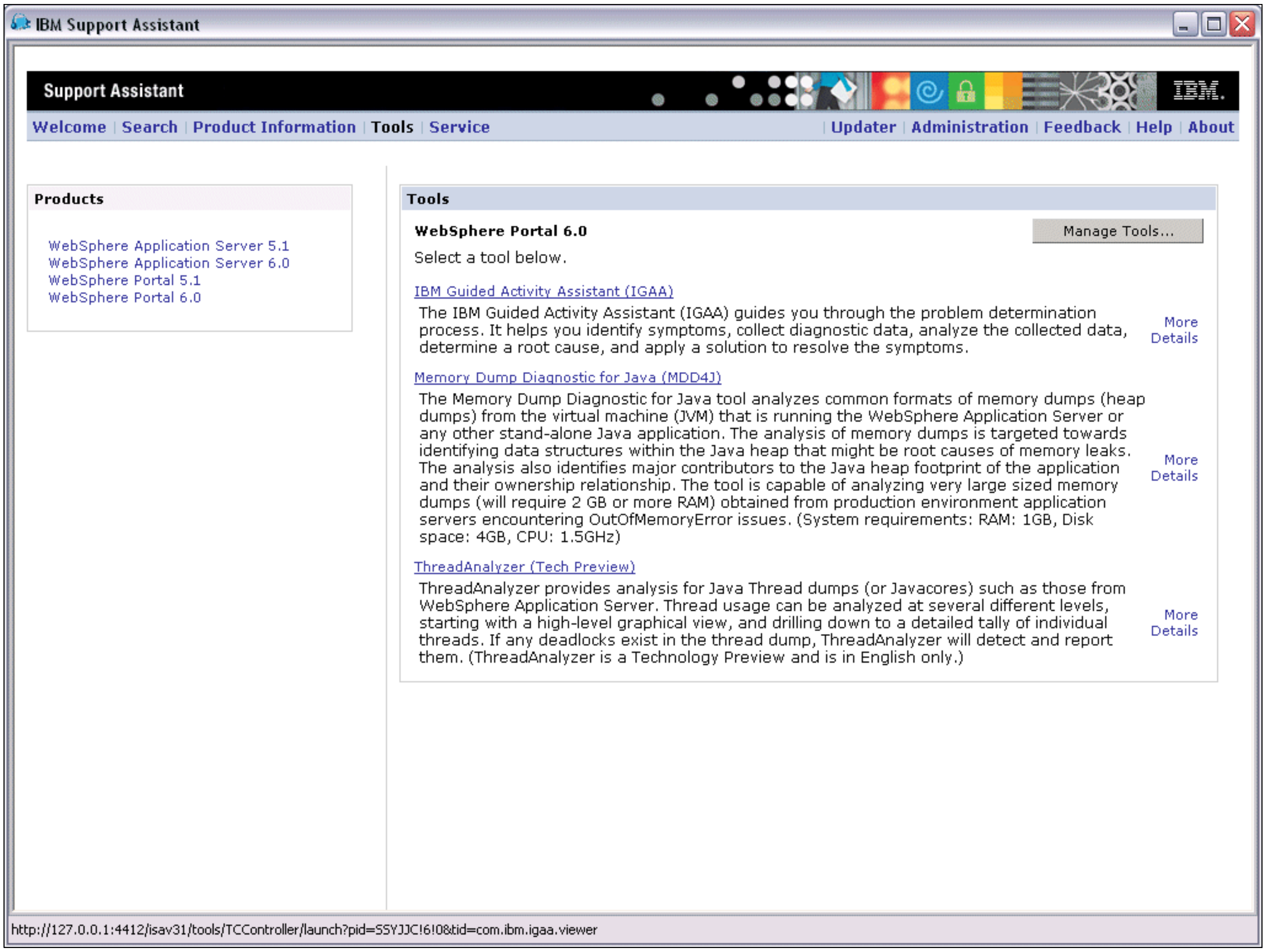
Now that you know what IGAA is at a high level and how it can help you, the next section will help give you an even better understanding of the value of IGAA with a real world example of how you can leverage IGAA to help solve problems.

How can I access the IBM Guided Activity Assistant (IGAA)

The IBM Guided Activity Assistant is unique in that it is solely delivered through the IBM Support Assistant (ISA). As previously documented in “Best practices” on page 174, we recommend installing the IGAA tool plug-in into ISA.

To launch IGAA, open ISA and go to the Tools feature and select the **IBM Guided Activity Assistant (IGAA)** tool. See Figure A-30.

Important: If you have not installed the IGAA tool plug-in, you will need to do so through the ISA Updater feature before you can access IGAA through ISA. For details about using the Updater feature through ISA, refer to “IBM Support Assistant (ISA)” on page 170.


IBM Support Assistant
Support Assistant
IBM.
Welcome | Search | Product Information | Tools | Service
Updater | Administration | Feedback | Help | About
Products
WebSphere Application Server 5.1
WebSphere Application Server 6.0
WebSphere Portal 5.1
WebSphere Portal 6.0
Tools
Manage Tools...
More Details
More Details
More Details

WebSphere Portal 6.0

Select a tool below.

IBM Guided Activity Assistant (IGAA)

The IBM Guided Activity Assistant (IGAA) guides you through the problem determination process. It helps you identify symptoms, collect diagnostic data, analyze the collected data, determine a root cause, and apply a solution to resolve the symptoms.

Memory Dump Diagnostic for Java (MDD4J)

The Memory Dump Diagnostic for Java tool analyzes common formats of memory dumps (heap dumps) from the virtual machine (JVM) that is running the WebSphere Application Server or any other stand-alone Java application. The analysis of memory dumps is targeted towards identifying data structures within the Java heap that might be root causes of memory leaks. The analysis also identifies major contributors to the Java heap footprint of the application and their ownership relationship. The tool is capable of analyzing very large sized memory dumps (will require 2 GB or more RAM) obtained from production environment application servers encountering OutOfMemoryError issues. (System requirements: RAM: 1GB, Disk space: 4GB, CPU: 1.5GHz)

ThreadAnalyzer (Tech Preview)

ThreadAnalyzer provides analysis for Java Thread dumps (or Javacores) such as those from WebSphere Application Server. Thread usage can be analyzed at several different levels, starting with a high-level graphical view, and drilling down to a detailed tally of individual threads. If any deadlocks exist in the thread dump, ThreadAnalyzer will detect and report them. (ThreadAnalyzer is a Technology Preview and is in English only.)

http://127.0.0.1:4412/isav31/tools/TCController/launch?pid=5SYJJC!6!0&tid=com.ibm.igaa.viewer

Figure A-30 Access IBM Guided Activity Assistant via IBM Support Assistant

Best practices

The most complete documentation on IGAA Best Practices can be found in this particularly useful document, “The Support Authority: Introducing the IBM Guided Activity Assistant”. This document can be found at:

`http://www.ibm.com/developerworks/websphere/techjournal/0705_supauth/0705_supauth.html`

B

Maintenance: Fix strategy, backup strategy, and migration strategy

This appendix discusses best practice approaches and procedures used during the maintenance phase of a WebSphere Portal deployment.

Backup strategy

A complete and thoroughly tested backup and recovery procedure is essential for any production environment. WebSphere Portal Server is no different. You should develop complete disaster recovery strategies and approaches and test those procedures in a testing environment. Once verified, you should implement these same procedures in the WebSphere Portal Server production environment.

This section does not detail a step-by-step process for backup. Rather, it provides insight into an approach for WebSphere Portal Server backup that you can implement into your existing disaster recovery procedures.

Overview of the backup process

The approach described in this section implements full system backups while maintaining 24x7 operations. The approach consists of stopping a percentage of the clustered nodes and then taking file system backups of the WebSphere Application Server and WebSphere Portal Sever root directories while the Portal nodes and node agents are stopped. The remaining clustered nodes continue to operate and maintain 24x7 operations.

After the backups are complete on the first group of Portal nodes, those nodes are brought back online in the cluster, another group of nodes are stopped, and the process is repeated. We recommend making backups while the nodes are down to avoid incomplete backups because of open files while the backups are taking place.

Note: Software tools may exist that allow complete backups to be made while files are open, but this section does not discuss these tools. If you wish to implement these types of tools, you can adjust the approach appropriately.

The general outline of the approach is:

- Stop a group of Portal nodes in the cluster.
- Make a file system backup of the stopped nodes.
- Start the nodes.
- Stop another group of nodes in the cluster.
- Make a file system backup of the stopped nodes.
- Start the nodes.
- Repeat this process until all nodes have been stopped and file system backups are taken of each node.
- Stop the Deployment Manager node.
- Make a file system backup of the Deployment Manager node.
- Make a database backup of all the databases associated with WebSphere Portal.
- Restart the Deployment Manager.

Important: XMLAccess does not play a part in our backup approach. XMLAccess is not a tool that is designed for full backup purposes. XMLAccess is a tool designed for deploying Portal artifacts from one Portal environment to another Portal environment. For example, you can use XMLAccess to move Portal artifacts from your staging environment into your production environment once the Portal configuration has been thoroughly tested in the staging environment.

While XMLAccess does have features that can play a role in some backup situations, you should not rely on an XMLAccess export in a disaster recovery scenario. Thus, we have left XMLAccess out of the discussion for WebSphere Portal disaster recovery to avoid giving a false impression of the tool's capabilities.

Our approach to backup

We recommend the following approach to backup:

1. Determine the time of day when the maintenance window takes place, preferably when the load on the cluster is the lowest.
2. Based on your environment, determine the fewest number of Portal nodes that are required to handle the load during this maintenance window.
3. Based on the length of your maintenance window and the minimum number of Portal nodes required to handle the load, determine the architecture of your backup procedure.

   For example, if you have a maintenance window of two hours for a 10 node cluster, you will need a minimum of three Portal nodes up to meet the average load requirements for this time period. If you assume that you can back up the file systems in 30 minutes, you can then break the backup into two sections. Bring down five Portal nodes, make backups, and then bring them back online. Then, take down the other five nodes and make backups. This is the quickest approach in a 24x7 environment, because you have divided your backup process into two sections. However, if you have a nine node cluster and the load requires six nodes to be up, then you will have to divide it into three sections. Depending on the speed of your backup process, you might need to extend the maintenance window in this situation.

   For this example, we divide the backups into two sections of five nodes each.
4. Stop the individual Portal application servers on nodes 1 through 5 using the Deployment Manager Administrative Console.

Note: Ensure that you take steps to stop all Web traffic to the nodes that will be undergoing maintenance before you stop the portal application servers.

5. Stop the node agents for nodes 1 through 5 using the Deployment Manager Administrative Console.
6. Make sure no servers are running on nodes 1 through 5 by using the **`serverStatus.sh/bat`** command or by checking for running Java processes.
7. Make file system backups on each node, 1 through 5, of the WebSphere Application Server and WebSphere Portal root directories.
8. Start node agents through the command line on nodes 1 through 5 after file system backups are complete.
9. Synchronize the nodes through the Deployment Manager Administrative Console.

10. Start the individual Portal Application Servers on nodes 1 through 5 through the Deployment Manager Administrative Console.
11. Stop the individual Portal Application Servers on nodes 6 through 10 using the Deployment Manager Administrative Console.
12. Stop the node agents for nodes 6 through 10 using the Deployment Manager Administrative Console.
13. Make sure no servers are running on nodes 6 through 10 by using the `serverStatus.sh/bat command` or by checking for running Java processes.
14. Make file system backups on each node, 6 through 10, of the WebSphere Portal and WebSphere Application Server root directories.
15. Start node agents through the command line on nodes 6 through 10 after file system backups are complete.
16. Synchronize the nodes through the Deployment Manager Administrative Console.
17. Start the individual Portal Application Servers on nodes 6 through 10 through the Deployment Manager Administrative Console.
18. Stop the Deployment Manager server through the command line.
19. Make file system backups on the Deployment Manager node of the WebSphere Deployment Manager root directory.
20. Make online database backups of the WebSphere Portal Server databases using the backup tools associated with the database server used in your environment.
21. Once file system backups and database backups are complete, start the Deployment Manager server from the command line.

Once again, these steps are not meant to provide a detailed step-by-step procedure but rather an approach to implementing a backup and recovery procedure for WebSphere Portal Server. You can automate many of these steps using scripts. Complete and reliable backups are critical. However, each backup plan is very specific to the environment. Thus, this general approach outlines the basic requirements for a full WebSphere Portal Server backup plan.

> **Recommendation:** Make backups after the initial install when WebSphere Portal Server is using Cloudscape and after you have configured WebSphere Portal Server to use the external database and LDAP servers. The point here is that it is a best practice to make backups before beginning a major configuration change so that in case of serious problems with the configuration you can fall back to the backup to restore the environment.

Some additional best practices

- Perform a backup after every major installation/configuration step. It may save time if problems occur and may avoid a complete rebuild.
- If you cannot perform a back after every major installation step because of time or resource constraints, then back up after the initial install and before federating into cells if clustering.
  - A Cloudscape install is a backup of last resort.
  - Back up prior to federation, because most problems happen during federation.
- Make backup copies of the wpconfig.properties file. In fact, make multiple copies and keep them in multiple places.
  - It takes time to configure the file correctly. Once done, you do not want to do it again.

- Make a copy after every major configuration (Database server, LDAP server, Web server, and so on).
- If all else fails, you can restore from the Cloudscape based backup, replace the default wpconfig.properties file, and rerun the configuration tasks. The entire process is scriptable.

Fix strategy

Maintenance is applying a service update to existing installed software, while upgrading to a newer version or release is covered in "Migration strategy" on page 219. This distinction is important.

IBM WebSphere Portal development and support teams strive to ensure that only the highest quality of code is made available to our customers, yet on occasion a problem will not be caught before its release. When this happens, an Authorized Program Analysis Report (APAR) is created to document the issue. The IBM Software Support Handbook (`http://techsupport.services.ibm.com/guides/handbook.html`) defines an APAR as:

"A formal report to IBM development, of a problem caused by a suspected defect in a current unaltered release of an IBM program. An APAR may also be used by development to document new function being delivered in the maintenance stream."

In the case where the APAR results in a fix or other corrective service, it is made available to customers in the form of Maintenance Delivery Vehicles (MDVs) that might be Interim Fixes, Fix Packs, Refresh Packs, or Manufacturing Refreshes. Generally, these are considered gradations of corrective service that is available, such that Fix Packs include an accumulation of Interim Fixes, Refresh Packs include everything that has gone into the Fix Packs they supersede, and so on.

Table B-1 defines the most commonly released MDVs.

Table B-1 Commonly released Maintenance Delivery Vehicles

MDV	Characteristics
Release	► A new WebSphere Portal that includes major new functionality, such as V6.0. ► This is a separate installation that can coexist with other WebSphere Portal releases. ► Full testing of all applications with a new release is recommended. ► Requires migration to upgrade; distributed through Passport Advantage.

MDV	Characteristics
Refresh Pack	► A package that may include new features and fixes, such as V6.0.1. ► Refresh Packs are cumulative, so V6.0.2* would include features and fixes contained in V6.0.1, as well as any subsequent Fix Packs and Interim Fixes published for V6.0.1. ► Refresh Packs uninstall all Fix Packs and Interim Fixes previously applied to the release. So, it is necessary to check the list of delivered fixes to determine if an Interim Fix must be reinstalled. ► Regression testing of critical functions with new Refresh Pack is strongly recommended to ensure your custom applications continue to function as that. ► A Refresh Pack which also provides new installation media (electronic or physical) is called a Manufacturing Refresh. Typically, a Manufacturing Refresh will have a Refresh Pack available as well for existing customers to install into their existing environment.
Fix Pack	► This is the standard delivery for updates; it has been fully regression tested by IBM prior to release. ► A Fix Pack is a cumulative package of only fixes, such as V6.0.1.1, unless otherwise noted. ► Fix Packs install on top of a specific Refresh Pack, such as applying V6.0.1.1 to V6.0.1.0, or onto the base level of code (6.0.0.0) in some cases. ► Fix Packs also install on top of a previous fix pack, such as applying V6.0.1.1 to V6.0.0.1. ► Fix Packs are cumulative, so V6.0.1.3* includes all fixes in V6.0.1.1. ► Fix Packs uninstall all Interim Fixes applied to the release since the last Refresh Pack or Fix Pack was installed. Therefore, it is necessary to check the list of delivered fixes to determine if an interim fix needs to be reinstalled. ► Testing of critical functions with the new fix pack is recommended to ensure your custom applications continue to function as expected.

MDV	Characteristics
Fix	► A single published emergency fix, such as PK29999. ► A fix is an Interim Fix, test fix or cumulative fix that resolves one or more product defects. ► A fix can be applied to a release, Refresh Pack, or Fix Pack where applicable ► Interim Fixes are created when a stand-alone fix is required between Fix Packs. They are validated by at least one customer and verified by IBM before publishing. Test fixes are not generally released but might be provided by IBM support during the course of a service call (PMR). ► We recommend that you test functions affected by the fixed WebSphere component. ► Reference the Recommended fixes and updates for WebSphere Portal page for currently available fixes at: http://www-1.ibm.com/support/docview.wss?rs=688&uid=swg27007603. When urgent, the fix will also be highlighted on the Support page.

This section describes two different approaches to help you avoid problems (maintenance) and correct or prevent problems (fix) in your environment.

Overview of the maintenance strategy

Among the most beneficial actions an administrator can perform on a portal is to keep it up to date with regard to service levels. By keeping the environment current, the likelihood of encountering a problem already corrected (“rediscovery”) is very rare, and allows the environment to take advantage of performance and stability improvements integrated into later service releases. Long gone are the days of installing some middleware on a server and sticking it in a closet with the expectation that it will not need any additional attention. Today’s environments can include a multitude of client and server platforms, dynamic content creation and presentation, and challenges to keep pace with changes in numerous industry standards.

The TechNote “Update Strategy for WebSphere Portal versions 5.1 and 6.0” (#1248245) describes the current philosophy of the development and support teams. In short, we deliver Fix Packs on a periodic schedule of release every three to six months. This TechNote can be found at:

http://www-1.ibm.com/support/docview.wss?rs=688&uid=swg21248245

Guidance is provided from IBM support showing the current list of recommendations for WebSphere Portal in the TechNote “Recommended fixes and updates for WebSphere Portal” (#7007603). Customers are recommended to use this as a foundation for understanding which service release level to use in their environment, and which Interim Fixes are considered to be critical in nature, to prevent rediscovering problems already corrected. This TechNote can be found at:

http://www-1.ibm.com/support/docview.wss?rs=688&uid=swg27007603

Installing maintenance on a clustered environment is different from installing it on a stand-alone node. Be sure to read and understand the installation instructions in the WebSphere Portal V6.0 Information Center regarding the installation of updates in a clustered environment, as covered in the topic “Performing upgrades in a 24x7 environment.” This document can be found at:

http://publib.boulder.ibm.com/infocenter/wpdoc/v6r0/index.jsp?topic=/com.ibm.wp.ent.doc/wpf/clus_upgrade.html

Our approach to maintenance

A well designed maintenance strategy will work from a “bottom up” philosophy, with an essentially duplicate copy of the production environment available for in-house quality assurance (QA) testing. For example, if the production environment runs on a clustered series of servers running IBM AIX, then a staging or QA environment should also have a cluster of servers on AIX.

From the bottom up, we have these aspects to consider when comparing the QA and production environments:

- Hardware (from disk arrays to network connectivity, memory, and CPU)
- Operating system (should be the same in QA as in production, even to the fix level)
- Application Server
- Database server
- Security server (LDAP, User registry, or even third-party external security manager, or ESM)
- Web server

And any other such service integrated into your business.

Many customers run their production environments using an “n-1” maintenance strategy, meaning that it does not always operate on the “latest and greatest” levels of software available, but the next most recent levels after being proven in their own QA systems.

When a new Fix Pack or other higher level MDV is available, it is installed on a QA environment to begin thorough testing within the local environment to ensure no problems are introduced by its installation. The upgrade can be installed, uninstalled, and reinstalled many times to effectively practice for the eventual upgrade to the production environment.

Fix Packs and Refresh Packs install into an existing environment with the Portal Update Installer (PUI) utility that is available with a command-line interface as well as a graphical interface.

Administrators familiar with the PUI (http://www.ibm.com/support/docview.wss?rs=688&uid=swg24006942) and its documentation in the version 6.0 Information Center (http://publib.boulder.ibm.com/infocenter/wpdoc/v6r0/index.jsp?topic=/com.ibm.wp.ent.doc/wpf/portalupdateinstaller.html) can save time by familiarizing themselves with the utility by practicing on a QA, development, or “sandbox” system before having to use it in an emergency.

Keeping thorough notes and documenting any pitfalls or lessons learned is key to success when applying new fixes to production, where limiting the impact to users is the primary objective. With enough practice, the maintenance window this might require could be shortened and in many cases even prevent any outage seen by customers.

After installing in QA, the system should be rigorously tested to ensure that no problems are introduced and that all custom content remains functional and compatible with the later releases. It is very important to monitor the system as well as the product support page for any late-breaking news regarding the MDV during this QA period.

Refer to the MDV's readme, release notes, and any subsequent TechNotes that may follow its release for details on any compatibility or integration issues discovered. If problems arise in your environment during QA, using the Self-Help resources described throughout this document will aid in your successful upgrade.

Overview of the fix strategy

Often, a fix (interim or cumulative fix) is installed to correct a condition encountered in your environment. If you are a heavy or frequent user of a particular component or function, you might have the flexibility available to you to keep up to date on all of the latest fixes for it, such as for the XML Configuration interface (XML Access) or WCM. Even if you have not encountered a problem in one of these areas but a fix is available, it may prevent you from incurring an outage or down time by applying the fix.

Sometimes a Fix Pack or other such higher level MDV is released but with companion Interim Fixes available at the same time. Typically, these are for problems discovered very late in the testing cycles prior to release, too late to be included in the final build of the product. These fixes will have had specific testing to ensure they correct the issue discovered, and are usually well documented on the product support page.

Installing fixes on a clustered environment is different from installing on a stand-alone node. Be sure to read and understand the installation instructions in the WebSphere Portal V6.0 Information Center regarding the installation of fixes in a clustered environment, as covered in the topic "Installing interim fixes on a cluster node", which can be found at:

http://publib.boulder.ibm.com/infocenter/wpdoc/v6r0/index.jsp?topic=/com.ibm.wp.ent.doc/wpf/clus_upgradeifixes.html

Our approach to fixes

If you report a defect to IBM support, an APAR is opened and a fix is prepared, and it is tested by IBM to ensure it works as intended. In some cases, such as a rare configuration or in an unusual integration with an application or custom content not available to IBM support, a test fix might be provided before the fix is made generally available. In these cases, it is imperative that you test the fix in your local environment to ensure that it adequately corrects the condition reported and that you provide feedback to IBM support. After confirming that the fix works, it will be packaged and officially released from the product support page.

Interim Fixes are commonly available to customers, and quite valuable to the self-help troubleshooter. If you encounter a problem in your environment, use the troubleshooting tools and methods discussed elsewhere in this document to determine the root cause of the problem. Use the product support page to find if a fix is available to correct the condition by searching on relevant keywords (error message codes or strings, or relevant configuration/application names), review any APARs found, and then retrieve the fix for that APAR.

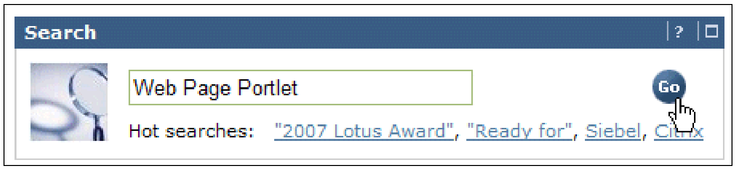
There is also a special type of Interim Fix that is sometimes made available, the Cumulative Fix, which typically includes fixes for a number of APARs. Functional areas currently offering these fixes include the WebSphere Member Manager (WMM), Java Content Repository (JCR) and Web Content Management (WCM) components, all made available through the product support page found at:

http://www.ibm.com/software/genservers/portal/support/

WebSphere Portal also ships with a variety of portlets available and installed for your use, if desired. Most of these portlets will make their updates available from the IBM WebSphere Portal Business Solutions catalog found at:

http://catalog.lotus.com/wps/portal/portal

The current versions of portlets, such as the Web Page or Web Clipping portlet, can be found by searching on the portlet's name on the catalog's main page, as shown in Figure B-1.

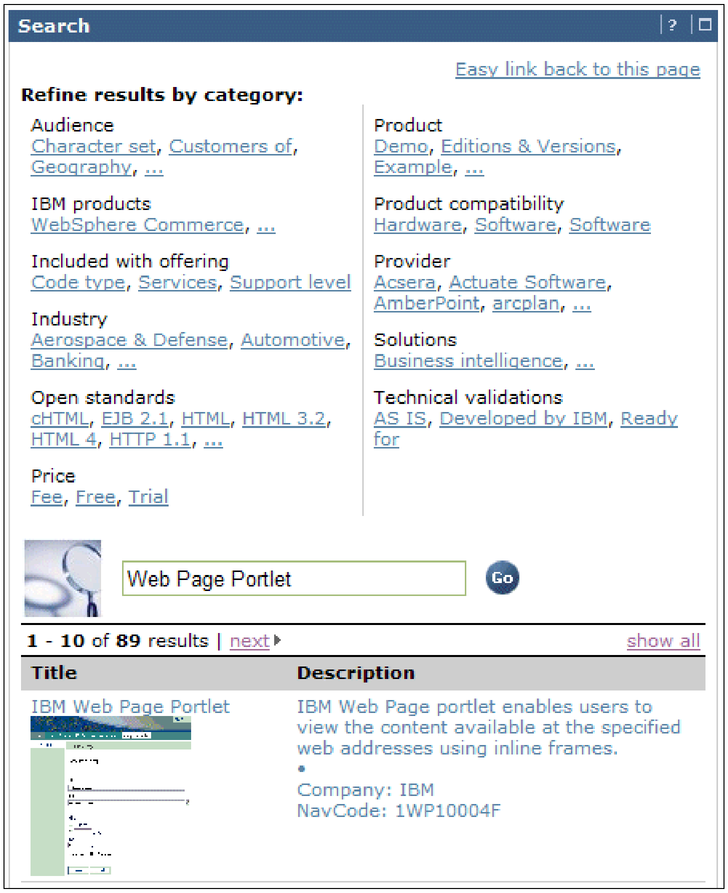


Web Page Portlet

Hot searches: "2007 Lotus Award", "Ready for", Siebel, Citrix

Figure B-1 Search box on the catalog

The search results should include the portlet's individual page from which you can download it and then "update" the deployed portlet using the portal's administrative interface, as shown in Figure B-2.



Easy link back to this page

Refine results by category:

Audience
Character set, Customers of, Geography, ...

IBM products
WebSphere Commerce, ...

Included with offering
Code type, Services, Support level

Industry
Aerospace & Defense, Automotive, Banking, ...

Open standards
cHTML, EJB 2.1, HTML, HTML 3.2, HTML 4, HTTP 1.1, ...

Price
Fee, Free, Trial

Product
Demo, Editions & Versions, Example, ...

Product compatibility
Hardware, Software, Software

Provider
Acsera, Actuate Software, AmberPoint, arcplan, ...

Solutions
Business intelligence, ...

Technical validations
AS IS, Developed by IBM, Ready for

Web Page Portlet

1 - 10 of 89 results | next ▸ show all

Title	**Description**
IBM Web Page Portlet	IBM Web Page portlet enables users to view the content available at the specified web addresses using inline frames. • Company: IBM NavCode: 1WP10004F

Figure B-2 Web Page Portlet search results on the catalog

Fixes are delivered through a variety of methods. Table B-2 on page 217 shows the normal distribution channels, by fix type.

Table B-2 Distribution channels by fix type

Fix type	Delivered only by IBM support to specific customers	Download linked from documentation on the product support page	Fix delivered by Fix Central[a]	Delivered from the portlet catalog	Integrated into later Fix Packs (or higher MDV)
Test fix	Yes	No	No	No	No
Interim fix	No	Yes	Yes[b]	No	Yes
Cumulative fix	No	Yes	Yes[b]	No	Yes
Business Portlet update	No	Yes	No	Yes	No (refer to the MDV's documentation for confirmation)

a. Fix Central (http://www.ibm.com/support/fixcentral) is a new service available from IBM support. Fixes available from Fix Central are identified by APAR number, Fix ID, version, platform, and other means, and the downloads can also include a fix's entire prerequisite chain (if it exists). This enhanced function improves the way we provide fixes and reduces the trouble we find when a fix requires another to be installed before it can be applied, and in many cases, that fix also requires others before it. When getting the fix from Fix Central, this complexity is handled by IBM by putting all of the required fixes together in one package for easy retrieval and installation.

b. WebSphere Portal is moving its entire fix repository over to Fix Central in the coming months, so until that is complete, we suggest that customers continue to use the product support page to locate and retrieve fixes to avoid having to look in multiple places. An announcement will be posted on the support page when the transition is complete.

When a fix is needed to correct a problem you have encountered, it rarely happens at a good time. IBM continues to invest resources into ensuring that such a fix is well tested, installs properly, and does what it is intended to do: fix the problem without introducing more problems. Many times there is urgency associated with the need for a fix especially if it is affecting a production environment. Nearly all of WebSphere Portal's test, interim, and cumulative fixes are installed with the Portal Update Installer (PUI) utility, which is available with a command-line interface as well as a graphical interface.

Administrators familiar with the PUI (http://www.ibm.com/support/docview.wss?rs=688&uid=swg24006942) and its documentation in Information Center Version 6.0 can save time by familiarizing themselves with the utility by practicing on a QA, development, or "sandbox" system before having to use it in an emergency.

There is a variation in this design for the WebSphere Portal Enable for z/OS® editions, since it does not use the Portal Update Installer or installation routines designed for the other distributed platforms. Updates and fixes for the z/OS platform (OS/390®) are released as ++ APARs (for emergency fixes) and PTFs, both of which are SMP/E installables and made generally available to those customers through their normal ordering and distribution channels. This section does not cover WebSphere Portal deployed on OS/390. It does, however, apply to the portal installed on the supported distributions of Linux on System z™ (SUSE and Red Hat), because the operating system is so similar to the distributions installed on x86 or RISC platforms and hardware.

Some additional best practices

Operate your production environment on as recent a service release as possible, and keep a mirror of your production environment available for testing upgrades and Interim Fixes before applying them to production.

Never “test” a fix or upgrade on a production environment! The only reasonable exception is when a critical problem exists solely on the production environment and cannot be reproduced on another system. Only then should an emergency fix be applied to the production environment to correct the specific problem.

Install IBM Support Assistant (see “IBM Support Assistant (ISA)” on page 170) on your administrative workstation and use it as your primary self-help interface, along with the product support page, the user forum, and other resources discussed in this Redpaper to identify fixes you need in your environment.

Keep a running list of the fixes, Fix Packs, and release levels of your software installed in a location off of your server(s) in case of catastrophe. To determine what you have installed on WebSphere Portal, refer to the TechNote “How to determine what fixes (or Fix Packs) are currently installed on WebSphere Portal” (#1246517) (http://www-1.ibm.com/support/docview.wss?rs=688&uid=swg21246517) and in the Portal Update Installer topic in Information Center Version 6.0, which is found at:

http://publib.boulder.ibm.com/infocenter/wpdoc/v6r0/index.jsp?topic=/com.ibm.wp.ent.doc/wpf/portalupdateinstaller.html

Keep backup copies of the fixes you have installed also off of the server to which they have been applied. Keeping these in a centrally stored location will make setting up a new or backup system easier and more quickly synchronized with the fix levels already in your current environment.

Maintain your operating system, WebSphere Application Server, Web server, database server, and other software that you use in your portal environment with the portal in mind. Upgrading one of these components to a later version may cause your portal to no longer function properly, so verify before upgrading in your test environment that it is likely to have a positive impact. The documents listed in Table B-3 will help you determine what is and is not recommended.

Table B-3 Supporting documents

Document name	Notes
WebSphere Portal Detailed System Requirements: http://www-1.ibm.com/support/docview.wss?rs=688&uid=swg27007791	Page listing the minimum service levels tested and required for specific WebSphere Portal releases.
WebSphere Portal Support Statement: http://publib.boulder.ibm.com/infocenter/wpdoc/v6r0/index.jsp?topic=/com.ibm.wp.ent.doc/wpf/inst_req_supt.html	Defines "supported" and "unsupported" categories of software to be used in conjunction with WebSphere Portal.
WebSphere Portal Support Statement Addendum - Unsupported Products: http://www-1.ibm.com/support/docview.wss?rs=688&uid=swg27007699	Lists the products and levels of software specifically not supported with WebSphere Portal.

Document name	Notes
Recommended fixes and updates for WebSphere Portal: http://www-1.ibm.com/support/docview.wss?rs=688&uid=swg27007603	Document describing IBM support's current recommendations for WebSphere Portal.

With these directions in mind, it is a best practice to keep your application server (and process server, if in use) at the current fix levels, as well as your portal. Be sure to check for any specific recommendations from those product support teams to ensure any compatibility issues or limitations are understood and accommodated.

If you identify a fix that is applicable to your environment and it is already integrated into a service release (for example, a Fix Pack) that is more recent than your currently deployed software level, it is generally a best practice to install the later Fix Pack rather than the Interim Fix. Fix Packs have been more thoroughly tested than individual Interim Fixes, and offer a greater level of confidence that the provided fixes will interoperate without issue. Again, first install the Fix Pack in a non-production, testing environment to ensure that it not only corrects the issue you are experiencing, but does not have unintended side effects on your unique environment and custom content.

If you have trouble installing a fix or Fix Pack, use the IBM Automated Problem Determination Tool (http://www-1.ibm.com/support/docview.wss?rs=688&uid=swg24008662) or the IBM Support Assistant's Remote Log Collector utility (http://www-306.ibm.com/software/support/isa/) to capture the diagnostic data and log files necessary to find the root cause of the problem. Appendix A, "Using IBM tools to find solutions and promote customer self-help" on page 169 contains the details about best practices for using the IBM Support Assistant.

Migration strategy

This section contains a very high level overview of the core WebSphere Portal Server migration process. See the Redpaper *IBM WebSphere Portal V6: Best Practices for Migrating from V5.1*, REDP-4227, for a more detailed explanation of the migration steps, found at:

http://www.redbooks.ibm.com/abstracts/redp4227.html

There is also a section in *IBM WebSphere Portal V6: Best Practices for Migrating from V5.1*, REDP-4227 regarding some additional components you may have used in your WebSphere Portal Server, such as WebSphere Content Manager and Personalization.

What is about to happen: a simple overview of the migration process

Migration is the process of reconstructing an existing IBM WebSphere Portal Server V5.1 environment on an IBM WebSphere Portal Server V6 environment so that the latter is identical to the former.

The following are the artifacts that are migrated in the process:

- Themes and Skins
- Pages
- Screens
- Portlet applications

- Access control
- User customization
- Virtual portals
- Markups
- Global settings
- Portal resources
- Workplace Web Content Manager content and components
- Document Manager content
- Personalization rules
- Credential vault slots

Note: For the rest of this section regarding migration, when referring to the WebSphere Portal Server V5.0 or V5.1 that you are migrating from, the document will refer to this server as your source server. When the document refers to your newly installed WebSphere Portal Server V6.0, it will refer to it as the target server.

The migration process first collects the files it needs from the source WebSphere Portal Server by using the Property Collector. Depending on what types of applications you have on your source system, there may be a few files that will need to be moved manually. After the Property Collector has been run, the next task is the export of the source server. This will create a xmlaccess file containing the information related to the source configuration. At this point in the migration, all the information from the source WebSphere Portal Server has been collected and the rest of the migration focuses on the target server.

The WebSphere Portal Server migration code will then perform a series of filters and translations to the XML file created based on the source servers configuration to create an XML file that will then be used to create a layout of the target server. An example of WebSphere Portal Server artifacts that will be filtered out from the source server are the old WebSphere Portal Server administration portlets.

The last part of the core migration is importing the edited XML file into the target WebSphere Portal Server. This will create the page layout and portlet preferences into the target WebSphere Portal Server.

Where do you start: planning and considerations

There is much pre-migration work that needs to be done in order to start the migration. First, you need to have a target server installed and configured to use the same type of database (DB2) and configured to use the same type of LDAP (or other third-party authentication tool). If, for example, the source WebSphere Portal Server is using IBM Tivoli Directory Server as the user repository in the source portal, you will need to configure the target server to use the same type of LDAP with the same user repository data. Note that this does not mean it needs to point to the same instance, just to an instance with the same user repository information.

Consideration: Migrating to a clustered or federated target server is not supported. The one exception is the case where WebSphere Portal Server is installed onto a WebSphere Process Server profile. In this case, the target can be federated but *not* clustered. During the migration, the nodeagent also needs to be turned off.

However, we highly recommend not installing WebSphere Portal Server with WebSphere Process Server, but instead use the WebSphere Process Server Client (the WebSphere Process Server will then be running on a remote WebSphere Application Server). This will improve performance and simplify the migration process.

For clarification, the source WebSphere Portal Server can certainly be a clustered configuration. The restriction above is speaking specifically about the target server.

Important: There is a cumulative fix for migration, PK48603, that is recommended before starting the migration. It is available at:

http://www-1.ibm.com/support/docview.wss?uid=swg1PK48603

Before beginning the migration, test that you can access and run the server, but do not make any customization to the WebSphere Portal Server at this time. The only apps that should be installed by you before migrations are any predeployed applications. These apps are installed using the WebSphere Application Server administration console and cannot be installed during the WebSphere Portal Server migration. The XML file to configure WebSphere Portal Server to use the predeployed portlet will also need to be run before migration. See the WebSphere Portal Server V6 InfoCenter link below for more information regarding pre-deployed portlets:

http://publib.boulder.ibm.com/infocenter/wpdoc/v6r0/topic/com.ibm.wp.ent.doc/wps/adprdplt.html

Is it working: verify the migration

After the core migration task is completed, you should be able to access the WebSphere Portal Server and see the same pages and portlets on the target system as were on the source. The following is a link to the IBM WebSphere Portal Server Version 6.0 InfoCenter that provides some general advice for migration verification:

http://publib.boulder.ibm.com/infocenter/wpdoc/v6r0/topic/com.ibm.wp.ent.doc/wpf/mig_verify.html

When a problem occurs: troubleshooting techniques to help identify the problem

The following log files are used during the migration to track the progress of the migration task and will display errors that occur:

```
<wp_root>/log/MigrationMessages.log
<wp_root>/log/MigrationTrace.log
```

The MigrationMessages.log file contains the message code and a translated message text describing the status of the migration as it runs. There are three types of messages: Error, Warning, and Informational. For some of the common failing points, the Error or Warning messages may contain an action to perform to correct the issue.

The MigrationTrace.log file contains a running list of each sub-task that the migration has run along with the time stamp that it occurred. This is helpful when mapping an error in either the WebSphere Portal Server log (systemout.log) or the MigrationMessages.log to a particular migration task. Search this file for the last BUILD FAILED to see at what point the latest migration ended.

Other files of interest can be found in the <wp_root>/work directory. This is were the migration stores the temporary files that are generated during the migration. These files can be helpful in showing what the tasks are trying to import into the V6 system. Unless stated in a TechNote or by IBM support, we do not recommend making changes to these files.

Problems with migration can be put into three broad categories: Export, Import, and Post Migration. We list some techniques to help locate the root cause of the errors encountered during each category in the following sections.

Export

- Xmlaccess export failure

  The migration export involved the migration code performing a full xmlaccess export on the source server. If there is an error during this export, the files to review are the MigrationMessages.log and the wps_<time>.log file from the source server. Often, errors with the export indicate that there may be some missing prerequisite fixes missing from the source/target servers.

- Export of user groups failure

  Another task that is performed during the export phase is the export of the groups. This is a very common point of failure that can have several causes. In this task, the WebSphere Portal Server exports the groups from the V5.1 system to create an XML file that will then be used to import the groups and the ACLs put on the groups into the target server. This step is only needed when you have assigned a manager role to a group to manage another group. This is not a very common set-p for most WebSphere Portal Servers. It is often best that if the migration fails on this task to skip it if it is not required.

  Common exceptions that occur during the export of the groups are SizeLimitExceededException or an error from the LDAP itself stating that it cannot export the number of records that is being requested. A TechNote that explains how to fix this issue, if it caused by the WMM size limit, can be found at:

  http://www-1.ibm.com/support/docview.wss?uid=swg21259963

Import

- Missing WAR files

  One of the manual steps in the migration of the portal is to move a copy of all the WAR files that exist in the source portal into the sharedApp directory on the target portal (the exception to this is pre-deployed apps). If this has not been done, the migration will fail when it tries to install the portlets into the target server. If you have PK48603 or your WebSphere Portal version is 6.0.1.1 or higher, the logs will provide a list of all WAR files that are missing. If you are not at this level of the portlet, the migration will fail at each missing WAR file, one at a time, with a nullpointer until the all of the WAR files are found.

- Anonymous user ACLs

  It is possible in IBM WebSphere Portal V5.1 to assign the anonymous virtual user a role on an item greater than USER. If this is done, the migration import will fail, because this is not possible in IBM WebSphere Portal V6.0 and above. You will need to remove the offending ACL from the source portal and rerun the export and restart the import.

- Missing users from the LDAP

  If users have been removed from the LDAP that the portal is using but not removed from the portal, these users will appear in the export and they will cause a problem when the migration tries to import them. Installing PK48603 will allow the migration to filter these users out, thus avoiding these issues during the migration. You can also use the following TechNote to remove the invalid users. Use the information in this TechNote if you are having issues with invalid users and cannot install PK48603. This will make changes to your source portal and should be done with caution. The TechNote is found at:

  `http://www-1.ibm.com/support/docview.wss?uid=swg21260104`

Post migration

- Themes and skins

  All efforts were made to make WebSphere Portal Server V6.0 backwards compatible with WebSphere Portal Server V5.1 themes and skins. There are some cases where previous WebSphere Portal versions themes/skins will need to be changed to run correctly on V6.0. It is a good practice to test your themes and skins on a WebSphere Portal Server V6.0 before putting a migration into production in order to know ahead of time if your theme will require changes. It is best to hold onto the file changes and add them after the migration finishes.

> **Note:** Using WebSphere Portal Server V5.1 themes and skins on a V6.0 server will not automatically give the themes the drag and drop feature. Manual steps are need to add this feature.

- Custom portlets

  As with themes and skins, most WebSphere Portal Server V5.1 portlets will run unchanged on WebSphere Portal Server V6.0. If the portlet does not work, then it might need to be loaded into Rational Application Developer (RAD) Version 7 and compiled against the WebSphere Portal Server V6.0 JARs. Also, if the portlet used any internal portal APIs that changed between the versions of WebSphere Portal, the portlet code will have to be rewritten to use the current APIs. Portlets written with the Struts Portlet Framework (SPF) should be updated to use the newest Struts code and updated into the target server after migration is completed. JSF portlets will also need to be imported into RAD V7 and exported in order for them to be packaged with the latest JSF code as well.

> **Note:** WebSphere Portal Server V5.0 themes, skins, and portlets are not expected to run without changes on WebSphere Portal Server V6.0. It is very likely that they will need to be changed to run on WebSphere Portal Server V6.0.

How to find a solution: using IBM Self-Help tools and support

Common migration troubleshooting techniques can be found at this link in the WebSphere Portal Server InfoCenter:

`http://publib.boulder.ibm.com/infocenter/wpdoc/v6r0/topic/com.ibm.wp.ent.doc/wpf/mig_tbl.html`

After locating the error code in the logs that caused the migration to stop, if the error itself does not have a recommendation to resolve the issue, the next place to check is the WebSphere Portal Server support Web site. Here you can enter the error code and see if there are any TechNotes describing the error you received. The support Web site is found at:

http://www-306.ibm.com/software/genservers/portal/support/

If there is not a TechNote about your issue or the TechNote resolves a different issue with a similar error code, then the next step is to contact WebSphere Portal Server Level 2 support. Before doing this task, it will speed the PMR resolution if you collect the WebSphere Portal Server V6 migration mustgather document before opening a PMR. You can collect the mustgather either by running the AutoPD tool or the through the IBM Support Assistant following the directions found at:

http://www-1.ibm.com/support/docview.wss?uid=swg21246134

Details about using the IBM Support Assistant are in Appendix A, “Using IBM tools to find solutions and promote customer self-help” on page 169.

What is next: typical next steps

After the migration is complete, you can continue with the setup of your WebSphere Portal Server as though it was a new WebSphere Portal Server configuration. The server can now be clusted and performance testing can start.

Related publications

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics covered in this paper.

IBM Redbooks publications

For information about ordering these publications, see “How to get IBM Redbooks publications” on page 225. Note that some of the documents referenced here may be available in softcopy only.

- *IBM WebSphere V4.0 Advanced Edition Security*, SG24-6520
- *IBM WebSphere Portal V6: Best Practices for Migrating from V5.1*, REDP-4227
- *WebSphere Portal V5.0 Production Deployment and Operations Guide*, SG24-6391
- *WebSphere Portal Version 6 Enterprise Scale Deployment Best Practices*, SG24-7387
- *WebSphere V3.5 Handbook*, SG24-6161

Other publications

These publications are also relevant as further information sources:

- *HACMP for AIX 5L V5.2 Administration and Troubleshooting Guide*, SC23-4862

How to get IBM Redbooks publications

You can search for, view, or download IBM Redbooks publications, Redpapers, TechNotes, draft publications and Additional materials, as well as order hardcopy IBM Redbooks publications, at this Web site:

ibm.com/redbooks

Help from IBM

IBM Support and downloads

ibm.com/support

IBM Global Services

ibm.com/services

IBM WebSphere Portal V6 Self Help Guide

Key recommendations for optimal configuration and use

Problem avoidance, determination, and resolution

Best practices for security and maintenance

This IBM Redpaper focuses on considerations for the optimal configuration and use of IBM WebSphere Portal Server. We provide you with the information you need to deploy and manage your WebSphere Portal infrastructure, with the goal of problem avoidance. However, if issues occur, the reader is introduced to the various tools and techniques for problem determination and problem solving, including obtaining and installing fixes, how to contact support, and what type of information you should provide before engagement.

This guide is a must have resource for IT architects and administrators throughout the life cycle of a WebSphere Portal environment, from conception and planning to use and maintenance.

REDP-4339-00